LARGE-ANGLE SLEWING MANEUVERS FOR FLEXIBLE SPACECRAFT

by

HON MING CHUN

S.B., Massachusetts Institute of Technology (1981)
S.M., Massachusetts Institute of Technology (1982)

Submitted to the Department of Aeronautics and Astronautics
in Partial Fulfillment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY IN AERONAUTICS AND ASTRONAUTICS

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 1986

Signature of Author _____
Department of Aeronautics and Astronautics, August 8, 1986

Certified by _____
Professor J. Karl Hedrick
Professor of Mechanical Engineering

Certified by _____
Dr. James D. Turner
Thesis Supervisor
Cambridge Research, A Division of Photon Research Associates, Inc.

Certified by _____
Professor Wallace E. VanderVelde
Chairman, Doctoral Thesis Committee
Professor of Aeronautics and Astronautics

Certified by _____
Professor Bruce K. Walker
Charles Stark Draper Assistant Professor of Aeronautics and Astronautics

Accepted by _____
Professor Harold Y. Wachman
Chairman, Department Graduate Committee

LARGE-ANGLE SLEWING MANEUVERS FOR FLEXIBLE SPACECRAFT

by
Hon Ming Chun

Submitted to the Department of Aeronautics and Astronautics
on August 8, 1986 in partial fulfillment of the requirements
for the Degree of Doctor of Philosophy in Aeronautics and Astronautics

Abstract

During the study of various control schemes for large-angle slewing maneuvers of flexible spacecraft, a new class of closed-form solutions for finite-time linear quadratic optimal control problems has been found. This class of closed-form solutions involves Potter's solution for the differential matrix Riccati equation, which assumes the form of a steady-state plus transient term. Illustrative examples are presented, and comparisons with alternative closed-form solutions involving the state transition matrix are made, which show that the new solutions are computationally efficient. The relationship between the two types of closed-form solutions is brought out by means of reducing subspace transformations.

Closed-form solutions are presented for the free-final-time neighboring extremal path problem when the plant is linear time-invariant. While the numerical results of previous authors have been extremely sensitive to numerical inaccuracies, the current study shows that slight numerical modifications to the feedback gains can greatly improve the system performance. An extension to nonlinear plants is shown.

Nonlinear three-axis slewing maneuvers are presented for flexible spacecraft. Attempts by previous authors for obtaining the open-loop flexible body solution have met with numerical difficulties, such as rank deficiency in a matrix to be inverted, and nonconvergence for large maneuver angles. The present approach obtains a frequency-shaped rigid body nominal solution using a continuation method. The elastic degrees of freedom are then controlled using a frequency-shaped perturbation feedback controller involving the flexible plant linearized about the rigid body nominal solution at regular intervals. The frequency shaping used in both the nominal solution and the perturbation feedback reduces the excitation to higher frequency residual modes. A modified Kalman filter is shown for estimating the plant states. Numerical results show that this approach is feasible.

Thesis Supervisor: Dr. J. Karl Hedrick
Title: Professor of Mechanical Engineering

Thesis Supervisor: Dr. James D. Turner
Title: Leader, Control and Dynamics Group
Cambridge Research

Thesis Supervisor:	Dr. Wallace E. Vander Velde
Title:	Professor of Aeronautics and Astronautics
Thesis Supervisor:	Dr. Bruce K. Walker
Title:	Charles Stark Draper Associate Professor of Aeronautics and Astronautics

ACKNOWLEDGEMENTS

I wish to thank Dr. James D. Turner for his many years of friendship and in teaching me in the art and science of engineering. Without his enthusiastic support and encouragement, this thesis could not have been completed. Many thanks are due to Prof. Wallace E. VanderVelde for his help and guidance throughout my doctoral program.

I am grateful to the Advanced Systems Department of Charles Stark Draper Laboratory, Inc. for financial support during the first two years of my doctoral program, and to Cambridge Research, A Division of Photon Research Associates for supporting me during the last year of my studies.

Many thanks are expressed to Karen Swiech for helping me to translate my programs from PL/I to FORTRAN; Dr. Keto Soosaar for his encouragement and support; Dr. Jer-Nan Juang for the administration of the NASA Langley contract under which much of this work was done; David Bristow and Janice Boreiko for their quick and expert typing.

I am greatly indebted to my parents who, as hardworking immigrants, instilled in me the virtues of learning and hard work. Most of all, I wish to thank my wife, Barbara Wan, for her much needed encouragement, and for enduring the hardships of being the wife of a graduate student. She also did all of the plots using our personal computer.

This report was prepared at Cambridge Research, A Division of Photon Research Associates, and supported by the National Aeronautics and Space Administration, Langley Research Center under Contract NAS1-18098. Publication of this thesis does not constitute approval by Cambridge Research or the National Aeronautics and Space Administration of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

I hereby assign my copyright of this thesis to Cambridge Research, A Division of Photon Research Associates, Cambridge, Massachusetts.

Hon Ming Chun

Permission is hereby granted by Cambridge Research, A Division of Photon Research Associates to Massachusetts Institute of Technology to reproduce any or all of this thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

The subject of large angle slewing maneuvers for flexible spacecraft has been a topic of great interest for many years, as seen by the large number of papers published [1,2,4,7,9,10,13,15-18,20-23,27-30, 36-38,47-52,59-61,68,82-84,86-100,102-106,113]. As spacecraft designs become larger in size, the inherent flexibility of the component structures and the flexibility at the joints result in a flexible spacecraft with low system natural frequencies. For large spacecraft, it is impractical to stiffen the structure by mechanical means, because this involves the addition of a great deal of mass to the structure, with accompanying increase in costs. Hence active control must be employed to reduce the detrimental effects of flexibility on the performance of the spacecraft.

The type of control problem considered in this thesis is the large angle slewing maneuver problem for flexible spacecraft. Many early papers have developed open-loop methods [1,9,10,17,18,29,38,52,59-61,86,89-91, 93,97-99,104] by assuming that the control can be obtained from either Pontryagin's principle or by assuming that the control can be expressed in terms of a Fourier Series. More recent papers have developed closed-loop methods [4,9,10,15-18,20-22,27,28,30,36,37,47-51,68,82-84,92,94-96,103, 104,113] which consider classical regulator, tracking, and terminal tracking formulations. Moreover, in an effort to minimize the deleterious effects of control spillover, a number of authors have proposed frequency-shaping techniques [2,4,36,37,47,50,59,66,86,95,101] which attempt to roll-off the high frequency content of the applied control.

In the course of studying different control formulations for applications to slewing control of flexible spacecraft, a new class of closed-form solutions for finite-time linear quadratic optimal control

problems has been found to be very useful. This class of solutions, which is based on Potter's solution [72,73] for the differential matrix Riccati equation, is discussed in Chapter 2, which summarizes the work done by the author and others in [19-21,47-49,94-96]. Comparisons are made with closed-form solutions obtained from the state transition matrix, and the relationship between the two types of solutions illustrated using reducing subspace transformations.

As further illustration of the use of the closed-form solutions, Chapter 3 presents closed-form analytic solutions for the perturbation feedback gains for generating neighboring optimal paths in the slewing control of flexible spacecraft. The perturbation feedback control scheme for generating neighboring extremal paths has been treated by many authors [6,11,14,57,70,74,85] on problems such as re-entry flight paths and low-thrust guidance. However, all results have been obtained by numerical techniques, and some of the results were highly sensitive to numerical inaccuracies. In Chapter 3, slight numerical modifications are shown for greatly enhancing the performance of this type of controller.

Chapter 4 deals with nonlinear three-axis slewing maneuvers for flexible spacecraft. The problem of nonlinear slewing maneuvers for rigid spacecraft has been treated by several authors [16,28,88,89,102,104-106]. The slewing problem for flexible spacecraft has also been treated recently [88,89], however, with limited success due to numerical difficulties. In Ref. [88], the elastic degrees of freedom are included in the state and costate variables, and a continuation process employed for solving the resulting two-point boundary-value problem, with the rigid body solution as starting guess. Numerical difficulties were encountered when a partition of the state transition matrix to be inverted had rank-deficiency. In Ref. [89], the effects of elasticity are treated as disturbances (modelled as Fourier series) acting on the rigid body solution. By iteratively modifying the rigid body solution, the resulting open-loop controller can effectively take into account the flexibility of the spacecraft. However, numerical results are limited to maneuvers of a few degrees (i.e. $< 5^{o}$). The method proposed in Chapter 4 continues the approach proposed in Ref. [22], and seeks the frequency-shaped rigid body

solution via a continuation method which uses a single-axis solution as the starting guess. Instead of obtaining an open-loop solution for the flexible spacecraft, a perturbation feedback control is employed to control the elastic degrees of freedom while the open-loop rigid body nominal torque profile is applied. The use of frequency-shaping minimizes the control spillover to the unmodelled degrees of freedom. A simple Kalman filter is modelled to further demonstrate the utility of the formulation.

CHAPTER 2

CLOSED-FORM SOLUTIONS FOR FINITE-TIME LINEAR-QUADRATIC OPTIMAL CONTROL PROBLEMS

2.1 Introduction

During the design and analysis phases of optimal control synthesis, state and control trajectories are often computed to help the control engineer evaluate the control design. The most straightforward and most widely practiced method of computing the state and control trajectories is by numerically integrating the governing differential equations. This may be costly for flexible space structures which may have many elastic degrees of freedom. The reason for the high cost is two-fold: first, the large number of elastic degrees of freedom requires a large number of states to be integrated; and second, since the highest frequency of the system to be simulated increases as the number of elastic modes is increased, the integration step-size must be decreased correspondingly. Thus, the computational cost of the simulation increases rapidly as more elastic degrees of freedom are included.

This chapter presents a new class of closed-form solutions for finite-time linear-quadratic optimal control problems when the plant is time-invariant. With a closed-form solution, one can compute the response of the entire system at any point in time. Thus, the engineer can compute the system response at any desired interval, independent of system frequency. (Of course, if one needs time-history plots with good resolution, the time interval does depend on system frequency.) In addition, numerical roundoff errors are greatly reduced, because the number of floating point operations[1] (flops) required for computing

---

[1] A floating point operation is more or less the amount of work needed to do a floating point add, a floating point multiply, and a little subscripting.

closed-form solutions is much less than for numerical integration. Sensitivity partials may also be computed easily when closed-form solutions are available [100].

Other forms of closed-form solutions exist. However, such solutions either suffer from numerical instability or require too much computation. With a numerically stable and efficient closed-form solution, one may envision the day when feedback gains may be computed on-orbit in real time. Whether the solutions proposed in this chapter can fulfill such a goal is of great interest, and is a subject for further research.

Although there are many different finite-time optimal control problems, their solutions can be written as differential equations of only a few basic forms. The closed-form solutions of these basic differential equations are presented in Section 2.2, and example applications are provided in Section 2.3 for illustration.

2.2 Closed-Form Solutions of Basic Differential Equations

The solution of finite-time linear-quadratic optimal control problems involves the solution of differential equations which may be classified into five basic types:

Type 1

$$\dot{P}(t) = - P(t)A - A^{T}P(t) + P(t)EP(t) - Q \ , \tag{2.2.1}$$

Type 2

$$\dot{X}_1(t) = - [A - EP(t)]^{T}X_1(t) + F_1(t) \ , \tag{2.2.2}$$

Type 3

$$\dot{X}_2(t) = \ [A - EP(t)]X_2(t) + F_2(t) \ , \tag{2.2.3}$$

Type 4

$$\dot{Y}_1(t) = X_{1a}^T(t)EX_{1b}(t) \ , \tag{2.2.4}$$

and Type 5

$$\dot{Y}_2(t) = X_{2a}^T(t)Z^{-1}(t)EZ^{-1}(t)X_{2b}(t) \ . \tag{2.2.5}$$

Type 1 represents the well-known differential matrix Riccati equation with constant coefficients. Its solution, P(t), couples into the differential equations of Type 2 and Type 3. The functions $X_{1a}(t)$ and $X_{1b}(t)$ are solutions of differential equations of Type 2, and the functions $X_{2a}(t)$ and $X_{2b}(t)$ are solutions of differential equations of Type 3. In the above equations, A, E, and Q are (n × n) constant matrices, and the variables have the following dimensions:

P(t)	(n × n)
$X_1(t)$	(n × p)
$X_2(t)$	(n × q)
$Y_1(t)$	(r × s)
$X_{1a}(t)$	(n × r)
$X_{1b}(t)$	(n × s)
$Y_2(t)$	(i × m)
$X_{2a}(t)$	(n × l)
$X_{2b}(t)$	(n × m) .

The matrices E, Q, and P(t) must be symmetric. The function $F_i(t)$ is a term representing the forcing functions for the $X_i(t)$ differential

equations, and accordingly has dimension $(n \times p)$ for $i = 1$ and $(n \times q)$ for $i = 2$. For scalar control problems, the five types of differential equations become scalar differential equations. However, for multivariable control problems, $P(t)$ is always a square matrix; $X_i(t)$ $(i=1,2)$ may represent either a matrix or a vector; and $Y_i(t)$ $(i=1,2)$ may be either a matrix, vector, or scalar.

2.2.1 Solution for Type 1 Differential Equations

The Type 1 differential equations are defined by

$$\dot{P}(t) = -P(t)A - A^T P(t) + P(t)EP(t) - Q . \tag{2.2.6}$$

The soluton of the above differential matrix Riccati equation is well-known, and, in fact, can be expressed in several different forms. One of the most useful forms of the solution is due to Potter [12,45,62,72,73,75], and is given by the sum of the steady-state solution (i.e. the solution to an algebraic Riccati equation) [3,12,45,62,71], and a transient term:

$$P(t) = P_{ss} + Z^{-1}(t) , \tag{2.2.7}$$

where

$$0 = P_{ss}A + A^T P_{ss} - P_{ss}E P_{ss} + Q .$$

In order to obtain the differential equation for $Z(t)$, one needs the following expression for the derivative of a matrix inverse:

$$\frac{d}{dt}[Z^{-1}(t)] = -Z^{-1}(t)\dot{Z}(t)Z^{-1}(t). \tag{2.2.8}$$

Substituting (2.2.7) into (2.2.6), and making use of (2.2.8), one obtains

$$\dot{Z}(t) = \bar{A}Z(t) + Z(t)\bar{A}^T - E , \tag{2.2.9}$$

where

$$\bar{A} = A - EP_{ss}$$

is the closed loop system dynamics matrix. The solution for Z(t) can be cast in either of the following two forms [24,45,80]:

$$Z(t) = Z_{ss} + e^{\bar{A}(t-t_o)}[Z(t_o)-Z_{ss}]e^{\bar{A}^T(t-t_o)}, \quad Z(t_o) = [P(t_o)-P_{ss}]^{-1};$$

or

$$Z(t) = Z_{ss} + e^{-\bar{A}(t_f-t)}[Z(t_f)-Z_{ss}]e^{-\bar{A}^T(t_f-t)}, \quad Z(t_f) = [P(t_f)-P_{ss}]^{-1}; \tag{2.2.10}$$

where Z_{ss} satisfies the algebraic Lyapunov equation [5,33,45,69]:

$$0 = \bar{A}Z_{ss} + Z_{ss}\bar{A}^T - E .$$

It can be shown that Z(t) and $Z^{-1}(t)$ exist for well-posed optimal control problems.

As shown in Sections 2.2.2 - 2.2.5, the symmetric matrix Z(t) plays a central role in the solution of differential equations of Types 2, 3, 4, and 5.

2.2.2 Solutions for Type 2 Differential Equations

The Type 2 differential equations are characterized by differential equations with time-varying coefficient matrices, where the coefficient matrices are functionally dependent upon the Type 1 equations. The general form for the Type 2 equations is given by

$$\dot{X}_1(t) = -[A - EP(t)]^T X_1(t) + F_1(t) . \tag{2.2.11}$$

The homogeneous solution of Type 2 differential equations can be found by assuming a solution of the form:

$$X_{1h}(t) = Z^{-1}(t)W(t) , \tag{2.2.12}$$

where $W(t)$ is unknown. Substituting (2.2.7) and (2.2.12) into the homogeneous form of (2.2.11) gives the differential equation for $W(t)$:

$$\dot{W}(t) = \bar{A}W(t) , \tag{2.2.13}$$

with the solution

$$W(t) = e^{\bar{A}(t-t_o)}W(t_o) , \quad W(t_o) = Z(t_o)X_1(t_o) . \tag{2.2.14}$$

The solution for the homogeneous differential equation of Type 2 can then be written in transition matrix form as

$$X_{1h}(t) = \Phi_1(t,t_o)X_1(t_o) , \tag{2.2.15}$$

where

$$\Phi_1(t,t_o) = Z^{-1}(t)e^{\bar{A}(t-t_o)}Z(t_o) .$$

The transition matrix satisfies the nonautonomous transition matrix differential equation:

$$\frac{d}{dt}[\Phi_1(t,t_o)] = -[A - EP(t)]^T\Phi_1(t,t_o) , \quad \Phi_1(t_o,t_o) = I ,$$

and also satisfies the semi-group and inversion properties:

$$\Phi_1(t_2,t_o) = \Phi_1(t_2,t_1)\Phi_1(t_1,t_o)$$

$$= Z^{-1}(t_2)e^{\bar{A}(t_2-t_1)}Z(t_1)Z^{-1}(t_1)e^{\bar{A}(t_1-t_o)}Z(t_o)$$

$$= Z^{-1}(t_2)e^{\bar{A}(t_2-t_o)}Z(t_o) ,$$

and

$$\Phi_1(t_o,t) = \Phi_1^{-1}(t,t_o)$$

$$= [Z^{-1}(t)e^{\bar{A}(t-t_o)}Z(t_o)]^{-1}$$

$$= Z^{-1}(t_o)e^{-\bar{A}(t-t_o)}[Z^{-1}(t)]^{-1}$$

$$= Z^{-1}(t_o)e^{\bar{A}(t_o-t)}Z(t) .$$

The particular solution of the inhomogeneous differential equation of Type 2 is found by using the general solution, (2.2.15), and replacing $X_1(t_o)$ by an unknown time-varying function in a variation of parameters approach:

$$X_1(t) = \Phi_1(t,t_o)G(t) , \quad G(t_o) = X_1(t_o) . \tag{2.2.16}$$

Substitution of (2.2.16) into (2.2.11) leads to the differential equation for G(t):

$$\dot{G}(t) = Z^{-1}(t_o)e^{-\bar{A}(t-t_o)}Z(t)F_1(t) , \tag{2.2.17}$$

which has the solution:

$$G(t) = Z^{-1}(t_o)\int_{t_o}^{t} e^{-\bar{A}(\tau-t_o)}Z(\tau)F_1(\tau)d\tau + G(t_o) . \tag{2.2.18}$$

From (2.2.16) and (2.2.18), the particular solution then becomes:

$$X_1(t) = Z^{-1}(t)e^{\bar{A}(t-t_o)}Z(t_o)X_1(t_o)$$

$$+ Z^{-1}(t)\int_{t_o}^{t} e^{\bar{A}(t-\tau)}Z(\tau)F_1(\tau)d\tau$$

$$= \Phi_1(t,t_o)X_1(t_o) + \int_{t_o}^{t}\Phi_1(t,\tau)F_1(\tau)d\tau \ . \qquad (2.2.19)$$

The integral term in (2.2.19) is easily obtained when $F_1(t)$ can be expressed in terms of exponential matrices. Applications for the closed-form solutions (2.2.15) and (2.2.19) are presented in Sections 2.3.2.2, 2.3.3.2, and 3.4.

2.2.3 Solutions for Type 3 Differential Equations

Like the Type 2 differential equations the Type 3 differential equations are characterized by differential equations with time-varying coefficient matrices, where the coefficient matrices are functionally dependent on the Type 1 equations. The general form for the Type 3 equations is given by

$$\dot{X}_2(t) = [A - EP(t)]X_2(t) + F_2(t) \ . \qquad (2.2.20)$$

The technique for solving differential equations of Type 3 closely parallels the solution procedure used for Type 2 equations. The general solution to the homogeneous differential equation is assumed to have the form:

$$X_{2h}(t) = Z(t)W(t) \ , \qquad (2.2.21)$$

where W(t) is unknown. Substitution of (2.2.21) into the homogeneous form of (2.2.20) leads to

$$\dot{W}(t) = -\bar{A}^T W(t) , \tag{2.2.22}$$

which has the solution

$$W(t) = e^{-\bar{A}^T(t-t_o)} W(t_o) , \quad W(t_o) = Z^{-1}(t_o)X_2(t_o) . \tag{2.2.23}$$

On substituting (2.2.23) into (2.2.21), the general solution of the homogeneous differential equation of Type 3 can be expressed in transition matrix form as

$$X_{2h}(t) = \Phi_2(t,t_o)X_2(t_o) , \tag{2.2.24}$$

where

$$\Phi_2(t,t_o) = Z(t)e^{-\bar{A}^T(t-t_o)}Z^{-1}(t_o) .$$

The above transition matrix satisfies the nonautonomous transition matrix differential equation:

$$\frac{d}{dt}[\Phi_2(t,t_o)] = [A - EP(t)]\Phi_2(t,t_o) , \quad \Phi_2(t_o,t_o) = I ,$$

and satisfies the semi-group and inversion properties:

$$\Phi_2(t_2,t_o) = \Phi_2(t_2,t_1)\Phi_2(t_1,t_o)$$

$$= Z(t_2)e^{-\bar{A}^T(t_2-t_1)}Z^{-1}(t_1)Z(t_1)e^{-\bar{A}^T(t_1-t_o)}Z^{-1}(t_o)$$

$$= Z(t_2)e^{-\bar{A}^T(t_2-t_o)}Z^{-1}(t_o) ,$$

and

$$\Phi_2(t_o,t) = \Phi_2^{-1}(t,t_o)$$

$$= [Z(t)e^{-\bar{A}^T(t-t_o)}Z^{-1}(t_o)]^{-1}$$

$$= [Z^{-1}(t_o)]^{-1}e^{\bar{A}^T(t-t_o)}Z^{-1}(t)$$

$$= Z(t_o)e^{-\bar{A}^T(t_o-t)}Z^{-1}(t) .$$

The particular solution of the inhomogeneous differential equation of Type 3 is obtained by replacing the constant $X_2(t_o)$ of the general solution (2.2.24) with an unknown time-varying function in a variation of parameters approach:

$$X_2(t) = \Phi_2(t,t_o)G(t) , \quad G(t_o) = X_2(t_o) . \tag{2.2.25}$$

Introducing (2.2.25) into (2.2.20), one obtains the following differential equation for G(t):

$$\dot{G}(t) = Z(t_o)e^{\bar{A}^T(t-t_o)}Z^{-1}(t)F_2(t) ,$$

leading to

$$G(t) = Z(t_o)\int_{t_o}^{t} e^{\bar{A}^T(\tau-t_o)}Z^{-1}(\tau)F_2(\tau)d\tau + G(t_o) . \tag{2.2.26}$$

On substituting (2.2.26) into (2.2.25), the particular solution for $X_2(t)$ becomes:

$$X_2(t) = Z(t)e^{-\bar{A}^T(t-t_o)}Z^{-1}(t_o)X_2(t_o) + Z(t)\int_{t_o}^{t} e^{-\bar{A}^T(t-\tau)}Z^{-1}(\tau)F_2(\tau)d\tau$$

$$= \Phi_2(t,t_o)X_2(t_o) + \int_{t_o}^{t}\Phi_2(t,\tau)F_2(\tau)d\tau \ . \qquad (2.2.27)$$

The integral term in (2.2.27) can be easily evaluated when $F_2(t)$ is expressed in terms of products of $Z(t)$ and exponential matrices. Applications using the closed-form solutions (2.2.24) and (2.2.27) are presented in Sections 2.3.1.3, 2.3.2.4, and 2.3.3.3.

2.2.4 Solutions for Type 4 Differential Equations

The Type 4 differential equations are characterized by products of Type 2 solutions. The general form for the Type 4 equation is given by

$$\dot{Y}_1(t) = X_{1a}^T(t)EX_{1b}(t) \ . \qquad (2.2.28)$$

First, let us solve for the case where the equations for $X_{1a}(t)$ and $X_{1b}(t)$ are homogeneous, i.e.,

$$\dot{X}_{1a}(t) = -[A - EP(t)]^T X_{1a}(t) \ , \qquad (n \times r) \quad (2.2.29)$$

and

$$\dot{X}_{1b}(t) = -[A - EP(t)]^T X_{1b}(t) \ . \qquad (n \times s) \quad (2.2.30)$$

Let us assume a solution of the form

$$Y_1(t) = X_{1a}^T(t)W(t)X_{1b}(t) + Y_{1c} \ , \qquad (r \times s) \quad (2.2.31)$$

where Y_{1c} is an $(r \times s)$ unknown constant, and $W(t)$ is an $(n \times n)$ unknown function. Substitution of (2.2.31) into (2.2.28) leads to

$$- X_{1a}^T[\bar{A} - EZ^{-1}]WX_{1b} + X_{1a}^T\dot{W}X_{1b} - X_{1a}^TW[\bar{A} - EZ^{-1}]^TX_{1b} = X_{1a}^TEX_{1b} . \tag{2.2.32}$$

Since $X_{1a}(t)$ and $X_{1b}(t)$ are independent, and assuming $W(t)$ is independent of $X_{1a}(t)$ and $X_{1b}(t)$, one must have

$$- [\bar{A} - EZ^{-1}(t)]W(t) + \dot{W}(t) - W(t)[\bar{A} - EZ^{-1}(t)]^T = E ,$$

or

$$\dot{W}(t) = \bar{A}W(t) + W(t)\bar{A}^T - EZ^{-1}(t)W(t) - W(t)Z^{-1}(t)E + E . \tag{2.2.33}$$

On examination of (2.2.33) and comparison with (2.2.9), one sees that the solution for $W(t)$ is $W(t) = Z(t)$, which is indeed independent of $X_{1a}(t)$ and $X_{1b}(t)$. The solution for $Y_1(t)$ can then be written as

$$Y_1(t) = X_{1a}^T(t)Z(t)X_{1b}(t) + Y_{1c} , \quad Y_{1c} = Y_1(t_o) - X_{1a}^T(t_o)Z(t_o)X_{1b}(t_o)$$

$$= X_{1a}^T(t_o)\Phi_1^T(t,t_o)Z(t)\Phi_1(t,t_o)X_{1b}(t_o) + Y_1(t_o) - X_{1a}^T(t_o)Z(t_o)X_{1b}(t_o) , \tag{2.2.34}$$

where $\Phi_1(t,t_o)$ is defined by (2.2.15).

Since the differential equation for $Y_1(t)$ represents a quadrature, one could also write the solution for $Y_1(t)$ as the integral of $\dot{Y}_1(t)$:

$$Y_1(t) = \int_{t_o}^{t} X_{1a}^T(\tau)EX_{1b}(\tau)d\tau + Y_1(t_o)$$

$$= X_{1a}^T(t_o)\int_{t_o}^{t}\Phi_1^T(\tau,t_o)E\Phi_1(\tau,t_o)d\tau X_{1b}(t_o) + Y_1(t_o) . \tag{2.2.35}$$

Comparison of (2.2.34) and (2.2.35) yields the identity

$$\Phi_1^T(t,t_o)Z(t)\Phi_1(t,t_o) = Z(t_o) + \int_{t_o}^{t} \Phi_1^T(\tau,t_o)E\Phi_1(\tau,t_o)d\tau \ . \quad (2.2.36)$$

While (2.2.34) and (2.2.35) both represent the same solution for $Y_1(t)$, (2.2.34) is more efficient for numerical computation.

Let us now solve for the case where the equations for $X_{1a}(t)$ and $X_{1b}(t)$ are both inhomogeneous, i.e.,

$$\dot{X}_{1a}(t) = -[A - EP(t)]^T X_{1a}(t) + F_{1a}(t) \ , \quad (2.2.37)$$

and

$$\dot{X}_{1b}(t) = -[A - EP(t)]^T X_{1b}(t) + F_{1b}(t) \ . \quad (2.2.38)$$

Using a variation of parameters approach, one replaces the constant terms in (2.2.34) by an unknown function G(t):

$$Y_1(t) = X_{1a}^T(t)Z(t)X_{1b}(t) + G(t) \ , \quad (2.2.39)$$

where

$$G(t_o) = Y_1(t_o) - X_{1a}^T(t_o)Z(t_o)X_{1b}(t_o) \ .$$

Substitution of (2.2.9), and (2.2.37) through (2.2.39) into (2.2.28) gives the differential equation for G(t):

$$\dot{G}(t) = -X_{1a}^T(t)Z(t)F_{1b}(t) - F_{1a}^T(t)Z(t)X_{1b}(t) \ , \quad (2.2.40)$$

which has the solution

$$G(t) = G(t_o) - \int_{t_o}^{t} X_{1a}^T(\tau)Z(\tau)F_{1b}(\tau)d\tau - \int_{t_o}^{t} F_{1a}^T(\tau)Z(\tau)X_{1b}(\tau)d\tau \ . \tag{2.2.41}$$

The complete solution for $Y_1(t)$ is then written as

$$Y_1(t) = X_{1a}^T(t)Z(t)X_{1b}(t) + Y_1(t_o) - X_{1a}^T(t_o)Z(t_o)X_{1b}(t_o) - \int_{t_o}^{t} X_{1a}^T(\tau)Z(\tau)F_{1b}(\tau)d\tau - \int_{t_o}^{t} F_{1a}^T(\tau)Z(\tau)X_{1b}(\tau)d\tau \ . \tag{2.2.42}$$

The integral terms in (2.2.42) are easily computed when $F_{1a}(t)$ and $F_{1b}(t)$ are functions of exponential matrices. Example applications of the closed-form solution (2.2.34) are presented in Sections 2.3.2.2 and 3.4.

2.2.5 Solutions for Type 5 Differential Equations

The Type 5 differential equations are characterized by products of Type 3 solutions and $Z^{-1}(t)$. The general form for the Type 5 equations is given by

$$\dot{Y}_2(t) = X_{2a}^T(t)Z^{-1}(t)EZ^{-1}(t)X_{2b}(t) \ . \tag{2.2.43}$$

We first consider the solution of Type 5 differential equations where $X_{2a}(t)$ and $X_{2b}(t)$ are both homogeneous, i.e.,

$$\dot{X}_{2a}(t) = [A - EP(t)]X_{2a}(t) \ , \qquad (n \times 1) \tag{2.2.44}$$

and

$$\dot{X}_{2b}(t) = [A - EP(t)]X_{2b}(t) \ . \qquad (n \times m) \tag{2.2.45}$$

Let us assume a solution of the form

$$Y_2(t) = X_{2a}^T(t)W(t)X_{2b}(t) + Y_{2c}(t) , \qquad (2.2.46)$$

where Y_{2c} is an $(1 \times m)$ unknown constant, and $W(t)$ is an $(n \times n)$ unknown function. Differentiation of (2.2.46), using (2.2.44) and (2.2.45), and substitution into (2.2.43) leads to

$$X_{2a}^T[\bar{A} - EZ^{-1}]^T W X_{2b} + X_{2a}^T \dot{W} X_{2b} + X_{2a}^T W[\bar{A} - EZ^{-1}]X_{2b} = X_{2a}^T Z^{-1} E Z^{-1} X_{2b} . \qquad (2.2.47)$$

Since $X_{2a}(t)$ and $X_{2b}(t)$ are independent, and assuming $W(t)$ is independent of $X_{2a}(t)$ and $X_{2b}(t)$, one must have

$$[\bar{A} - EZ^{-1}(t)]^T W(t) + \dot{W}(t) + W(t)[\bar{A} - EZ^{-1}] = Z^{-1}(t)EZ^{-1}(t) ,$$

or

$$\dot{W}(t) = - W(t)\bar{A} - \bar{A}^T W(t) + Z^{-1}(t)EW(t) + W(t)EZ^{-1}(t) + Z^{-1}(t)EZ^{-1}(t) . \qquad (2.2.48)$$

On examination of (2.2.48) one observes that $W(t) = - Z^{-1}(t)$ is the solution for $W(t)$, which is indeed independent of $X_{2a}(t)$ and $X_{2b}(t)$. The general solution for $Y_2(t)$ is then

$$Y_2(t) = - X_{2a}^T(t)Z^{-1}(t)X_{2b}(t) + Y_{2c} , \quad Y_{2c} = Y_2(t_o) + X_{2a}^T(t_o)Z^{-1}(t_o)X_{2b}(t_o)$$

$$= - X_{2a}^T(t_o)\Phi_2^T(t,t_o)Z^{-1}(t)\Phi_2(t,t_o)X_{2b}(t_o)$$

$$+ Y_2(t_o) + X_{2a}^T(t_o)Z^{-1}(t_o)X_{2b}(t_o) , \qquad (2.2.49)$$

where $\Phi_2(t,t_o)$ is defined by (2.2.24).

An alternative solution for $Y_2(t)$ is obtained by forming the integral of $\dot{Y}_2(t)$, which is

$$Y_2(t) = \int_{t_o}^{t} X_{2a}^T(\tau)Z^{-1}(\tau)EZ^{-1}(\tau)X_{2b}(\tau)d\tau + Y_2(t_o)$$

$$= X_{2a}^T(t_o) \int_{t_o}^{t} \Phi_2^T(\tau,t_o)Z^{-1}(\tau)EZ^{-1}(\tau)\Phi_2(\tau,t_o)d\tau X_{2b}(t_o) + Y_2(t_o) . \quad (2.2.50)$$

Comparing (2.2.49) and (2.2.50), and observing that $X_{2a}(t_o)$ and $X_{2b}(t_o)$ are independent, one can write the identity

$$\Phi_2^T(t,t_o)Z^{-1}(t)\Phi_2(t,t_o) = Z^{-1}(t_o) - \int_{t_o}^{t} \Phi_2^T(\tau,t_o)Z^{-1}(\tau)EZ^{-1}(\tau)\Phi_2(\tau,t_o)d\tau . \quad (2.2.51)$$

Let us now solve for the case where the differential equations for $X_{2a}(t)$ and $X_{2b}(t)$ are both inhomogeneous:

$$\dot{X}_{2a}(t) = [A - EP(t)]X_{2a}(t) + F_{2a}(t) , \quad (2.2.52)$$

and

$$\dot{X}_{2b}(t) = [A - EP(t)]X_{2b}(t) + F_{2b}(t) . \quad (2.2.53)$$

Using a variation of parameters approach, one replaces the constant terms in (2.2.49) by an unknown function G(t), leading to:

$$Y_2(t) = - X_{2a}^T(t)Z^{-1}(t)X_{2b}(t) + G(t) , \quad (2.2.54)$$

where

$$G(t_o) = Y_2(t_o) + X_{2a}^T(t_o)Z^{-1}(t_o)X_{2b}(t_o) .$$

Substitution of (2.2.8), (2.2.9), and (2.2.52) through (2.2.54) into (2.2.43) gives the differential equation for G(t):

$$\dot{G}(t) = X_{2a}^T(t)Z^{-1}(t)F_{2b}(t) + F_{2a}^T(t)Z^{-1}(t)X_{2b}(t) , \qquad (2.2.55)$$

which has the solution

$$G(t) = G(t_o) + \int_{t_o}^{t} X_{2a}^T(\tau)Z^{-1}(\tau)F_{2b}(\tau)d\tau + \int_{t_o}^{t} F_{2a}^T(\tau)Z^{-1}(\tau)X_{2b}(\tau)d\tau . \qquad (2.2.56)$$

From (2.2.54) and (2.2.56), the complete solution for $Y_2(t)$ is

$$Y_2(t) = - X_{2a}^T(t)Z^{-1}(t)X_{2b}(t) + Y_2(t_o) + X_{2a}^T(t_o)Z^{-1}(t_o)X_{2b}(t_o) + \int_{t_o}^{t} X_{2a}^T(\tau)Z^{-1}(\tau)F_{2b}(\tau)d\tau + \int_{t_o}^{t} F_{2a}^T(\tau)Z^{-1}(\tau)X_{2b}(\tau)d\tau . \qquad (2.2.57)$$

The integral terms in (2.2.57) are easily computed when $F_{2a}(t)$ and $F_{2b}(t)$ can be expressed as products of Z(t) and exponential matrices. Differential equations of Type 5 do not appear often. However, closely related equations are shown in Sections 2.3.2.4 and 2.3.3.3.

Throughout the developments of this section, one can observe the close relationship between equations of Type 2 and Type 3, and also between equations of Type 4 and Type 5. Indeed, equations of Type 2 and Type 3 are formal adjoints of one another. An interesting property of this is the fact that the inner products of homogeneous solutions of Type 2 and Type 3 are constant. This can be shown by letting

$$Y(t) = X_1^T(t)X_2(t) , \qquad (2.2.58)$$

and forming the derivative of Y(t):

$$\dot{Y}(t) = -X_1^T(t)[A - EP(t)]X_2(t) + X_1^T(t)[A - EP(t)]X_2(t)$$

$$= 0 . \qquad (2.2.59)$$

This result is true for any $X_1(t)$ and $X_2(t)$ which satisfy homogeneous differential equations of Type 2 and Type 3, respectively.

Tables 2-1 and 2-2 provide a summary of the five basic differential equations and their solutions.

2.3 Example Applications of Closed-Form Solutions

This section presents solutions of finite-time linear-quadratic optimal control problems to illustrate the application of the solution forms shown in Section 2.2. The closed-form solutions for the linear optimal regulator are presented in Section 2.3.1. Section 2.3.2 presents closed-form solutions for the linear optimal controller with terminal constraints, and Section 2.3.3 treats the optimal deterministic disturbance accommodating/tracking controller. In all three cases, a comparison with the closed-form solution based on the open-loop solution is presented.

2.3.1 Linear Optimal Regulator

2.3.1.1 Optimal Control Problem and Necessary Conditions

The linear optimal regulator is the problem of minimizing the performance index

$$J = \frac{1}{2}\|y(t_f)\|^2_{P'} + \frac{1}{2}\int_{t_o}^{t_f} [\|y(t)\|^2_{Q'} + \|u(t)\|^2_R] \, dt , \qquad (2.3.1)$$

subject to the time-invariant system dynamics

$$\dot{x}(t) = Ax(t) + Bu(t) , \quad x(t_o) \text{ given}, \qquad (2.3.2)$$

Table 2-1. Summary of Basic Differential Equations and Their Solutions

Differential Equation	Solution
Type 1 $\dot{P}(t) = -P(t)A - A^{T}P(t) + P(t)EP(t) - Q$	$P(t) = P_{ss} + Z^{-1}(t)$
Type 2 $\dot{X}_1(t) = -[A - EP(t)]^{T}X_1(t) + F_1(t)$	$X_1(t) = \phi_1(t,t_o)X_1(t_o) + \int_{t_o}^{t} \phi_1(t,\tau)F_1(\tau)d\tau$
Type 3 $\dot{X}_2(t) = [A - EP(t)]X_2(t) + F_2(t)$	$X_2(t) = \phi_2(t,t_o)X_2(t_o) + \int_{t_o}^{t} \phi_2(t,\tau)F_2(\tau)d\tau$
Type 4 $\dot{Y}_1(t) = X_{1a}^{T}(t)EX_{1b}(t)$	$Y_1(t) = X_{1a}^{T}(t)Z(t)X_{1b}(t) + Y_1(t_o) - X_{1a}^{T}(t_o)Z(t_o)X_{1b}(t_o) - \int_{t_o}^{t} X_{1a}^{T}(\tau)Z(\tau)F_{1b}(\tau)d\tau - \int_{t_o}^{t} F_{1a}^{T}(\tau)Z(\tau)X_{1b}(\tau)d\tau$
Type 5 $\dot{Y}_2(t) = X_{2a}^{T}(t)Z^{-1}(t)EZ^{-1}(t)X_{2b}(t)$	$Y_2(t) = -X_{2a}^{T}(t)Z^{-1}(t)X_{2b}(t) + Y_2(t_o) + X_{2a}^{T}(t_o)Z^{-1}(t_o)X_{2b}(t_o) + \int_{t_o}^{t} X_{2a}^{T}(\tau)Z^{-1}(\tau)F_{2b}(\tau)d\tau + \int_{t_o}^{t} F_{2a}^{T}(\tau)Z^{-1}(\tau)X_{2b}(\tau)d\tau$

Table 2-2. Summary of Auxiliary Differential Equations and Their Solutions

Differential Equation	Solution
$\dot{Z}(t) = \bar{A}Z(t) + Z(t)\bar{A}^T - E$ $\bar{A} = A - EP_{ss}$ $Z(t_o) = [P(t_o) - P_{ss}]^{-1}, \quad Z(t_f) = [P(t_f) - P_{ss}]^{-1}$	$Z(t) = Z_{ss} + e^{\bar{A}(t-t_o)}[Z(t_o) - Z_{ss}]e^{\bar{A}^T(t-t_o)}$ $= Z_{ss} + e^{-\bar{A}(t_f-t)}[Z(t_f) - Z_{ss}]e^{-\bar{A}^T(t_f-t)}$
$\frac{d}{dt}[\Phi_1(t,t_o)] = -[A - EP(t)]^T\Phi_1(t,t_o)$ $\Phi_1(t_o,t_o) = I$	$\Phi_1(t,t_o) = Z^{-1}(t)e^{\bar{A}(t-t_o)}Z(t_o)$
$\frac{d}{dt}[\Phi_2(t,t_o)] = [A - EP(t)]\Phi_2(t,t_o)$ $\Phi_2(t_o,t_o) = I$	$\Phi_2(t,t_o) = Z(t)e^{-\bar{A}^T(t-t_o)}Z^{-1}(t_o)$

and

$$y(t) = Cx(t) \ , \tag{2.3.3}$$

where $\|x\|_W^2 = x^T W x$, y(t) is the m-dimensional output state, x(t) is the n-dimensional state vector, u(t) is the p-dimensional control vector, A is the (n × n) system dynamics matrix, B is the (n × p) control influence matrix, C is the (m × n) measurement influence matrix, $P_f = P_f^T = C^T P' C \geq 0$ is the terminal state weight matrix, $Q = Q^T = C^T Q' C \geq 0$ is the state weight matrix, and $R = R^T > 0$ is the control weight matrix. Figure 2-1 shows the block diagram for the feedback solution assuming perfect plant estimates. The necessary conditions for the minimization of the performance index can be shown to be [3,14]:

$$\dot{x}(t) = Ax(t) + Bu(t) \ , \qquad x(t_o) = x_o \ , \tag{2.3.4}$$

$$\dot{\lambda}(t) = - Qx(t) - A^T \lambda(t) \ , \quad \lambda(t_f) = P_f x(t_f) \ , \tag{2.3.5}$$

$$u(t) = - R^{-1} B^T \lambda(t) \ , \tag{2.3.6}$$

where $\lambda(t)$ is the costate or adjoint variable. The modified state equation is obtained by substituting (2.3.6) into (2.3.4), yielding

$$\dot{x}(t) = Ax(t) - BR^{-1} B^T \lambda(t) \ . \tag{2.3.7}$$

The closed-loop control is obtained by assuming a solution for $\lambda(t)$ of the form

$$\lambda(t) = P(t)x(t) \ . \tag{2.3.8}$$

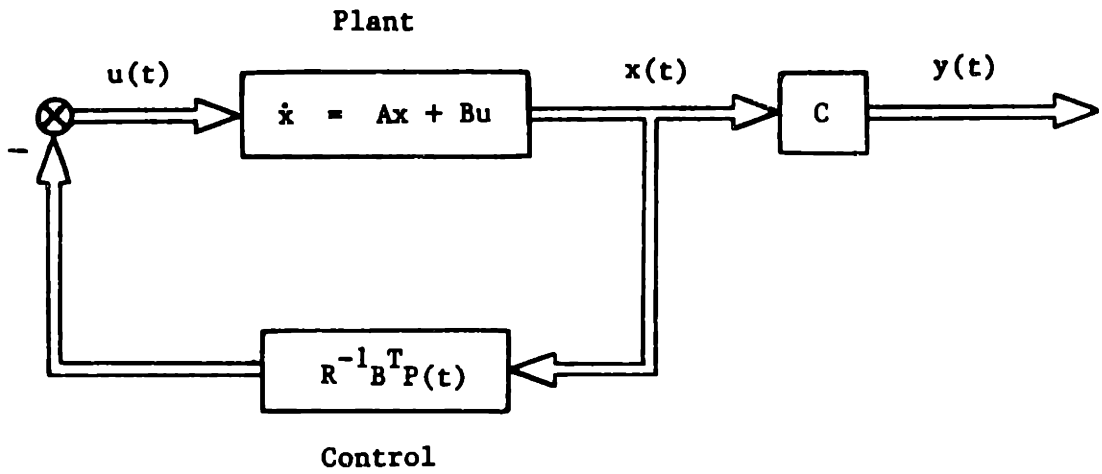


Figure 2-1. Block Diagram for the Deterministic Linear Optimal Regulator

Substituting (2.3.8) into (2.3.5) and (2.3.7) leads to

$$\dot{P}(t) = -P(t)A - A^T P(t) + P(t)EP(t) - Q \,, \quad P(t_f) = P_f \,, \tag{2.3.9}$$

where

$$E = BR^{-1}B^T \,.$$

Equation (2.3.9) is easily identified as the differential matrix Riccati equation or a Type 1 equation, with the solution

$$P(t) = P_{ss} + Z^{-1}(t) \,, \tag{2.3.10}$$

where

$$Z(t) = Z_{ss} + e^{-\bar{A}(t_f-t)}[Z(t_f)-Z_{ss}]e^{-\bar{A}^T(t_f-t)} \,, \quad Z(t_f) = [P_f-P_{ss}]^{-1},$$

or

$$Z(t) = Z_{ss} + e^{\bar{A}(t-t_o)}[Z(t_o) - Z_{ss}]e^{\bar{A}^T(t-t_o)} \,,$$

with

$$Z(t_o) = Z_{ss} + e^{-\bar{A}(t_f-t_o)}[Z(t_f) - Z_{ss}]e^{-\bar{A}^T(t_f-t_o)} \,.$$

2.3.1.2 Propagation of the Riccati Solution

The solution for Z(t) in (2.3.10) can be recursively generated at intervals of Δt by the following difference equation:

$$Z(t+\Delta t) = Z_{ss} + e^{\bar{A}\Delta t}[Z(t) - Z_{ss}]e^{\bar{A}^T\Delta t}$$

$$= C + e^{\bar{A}\Delta t}Z(t)e^{\bar{A}^T\Delta t} , \qquad (2.3.11)$$

where

$$C = Z_{ss} - e^{\bar{A}\Delta t}Z_{ss}e^{\bar{A}^T\Delta t} .$$

Equation (2.3.11) requires roughly $3n^3/2$ flops to propagate the symmetric Z(t) matrix. In computing the Riccati solution of (2.3.10), the symmetric definite matrix inversion requires $n^3/2$ flops. (The matrix Z(t) is positive definite if $P_f - P_{ss} > 0$, negative definite if $P_f - P_{ss} < 0$.) Thus, a total of $2n^3$ flops are required to compute P(t) explicitly (See Appendix A for a detailed operation count). During computer simulation studies, Z(t) and P(t) need not be computed explicitly if one is only interested in the state and control trajectories. In the following development, it is shown that propagation of the states and controls requires only $O(n^2)$ flops, compared with the propagation of Z(t) and P(t), which require $O(n^3)$ flops. Thus, the avoidance of Z(t) and P(t) propagation is of great practical value in reducing the computational requirements for the simulation of the system response.

An alternative way of computing the matrix Riccati solution is the well-known Kalman-Englar method [54], where P(t) is propagated at intervals of Δt by

$$P(t+\Delta t) = [\Theta_{21}(t+\Delta t,t) + \Theta_{22}(t+\Delta t,t)P(t)][\Theta_{11}(t+\Delta t,t) + \Theta_{12}(t+\Delta t,t)P(t)]^{-1} , \qquad (2.3.12)$$

where $\Theta_{ij}(t+\Delta t,t)$ are partitions of the transition matrix for the state-costate system, i.e.,

$$\frac{d}{dt}[\Theta(t,t_o)] = \Omega\Theta(t,t_o) , \quad \Theta(t_o,t_o) = I , \tag{2.3.13}$$

and

$$\Omega = \begin{bmatrix} A & -E \\ -Q & -A^T \end{bmatrix} .$$

The solution for the initial Riccati matrix is obtained by writing

$$P(t_f) = [\Theta_{21}(t_f,t_o)+\Theta_{22}(t_f,t_o)P(t_o)][\Theta_{11}(t_f,t_o) + \Theta_{12}(t_f,t_o)\, P(t_o)]^{-1} , \tag{2.3.14}$$

and solving for $P(t_o)$:

$$P(t_o) = [\Theta_{22}(t_f,t_o) - P(t_f)\Theta_{12}(t_f,t_o)]^{-1}[P(t_f)\Theta_{11}(t_f,t_o) - \Theta_{21}(t_f,t_o)] . \tag{2.3.15}$$

The number of operations required for the propagation of P(t) via (2.3.12) is n^3 flops for each of the $\Theta_{22}P$ and $\Theta_{12}P$ products, $n^3/3$ flops for the L-U decomposition (with partial pivoting) of the $[\Theta_{11} + \Theta_{12}P]$ term, $n^3/2$ for forward elimination, and $n^3/6$ for back-substitution, where symmetry of P(t) is taken into account. The total number of operations adds up to about $3n^3$ flops. Thus, the Kalman-Englar method requires about 50% more operations than Potter's method for propagating P(t) at intervals of Δt. Moreover, numerical difficulties arise in the Kalman-Englar method when Δt is chosen too large, causing the term to be inverted to be nearly singular [58]. For Potter's method, such difficulties do not occur. Another solution approach is the negative exponential solution derived by Vaughan [109] which is a numerically stable algorithm. However, since this method involves complex eigenvectors of the Hamiltonian matrix, the use of complex arithmetic causes the operation count to be many times higher than that for the Kalman-Englar method.

2.3.1.3 State Trajectory Solution

To determine the closed-loop system response under the assumptions of perfect plant knowledge and perfect state estimation, (2.3.6) and (2.3.8) are introduced into (2.3.4), leading to

$$\dot{x}(t) = [A - EP(t)]x(t) , \quad x(t_o) = x_o , \tag{2.3.16}$$

which is a homogeneous form of the Type 3 differential equation. From (2.2.24), the solution for x(t) is

$$x(t) = \Phi_2(t,t_o)x_o$$

$$= Z(t)e^{-\bar{A}^T(t-t_o)}Z^{-1}(t_o)x_o . \tag{2.3.17}$$

Substitution of the solution for Z(t) from (2.3.10) into (2.3.17) yields

$$x(t) = \{Z_{ss} + e^{\bar{A}(t-t_o)}[Z(t_o)-Z_{ss}]e^{\bar{A}^T(t-t_o)}\}e^{-\bar{A}^T(t-t_o)}Z^{-1}(t_o)x_o$$

$$= Z_{ss}a(t) + b(t) , \tag{2.3.18}$$

where

$$a(t) = e^{-\bar{A}^T(t-t_o)}Z^{-1}(t_o)x_o ,$$

and

$$b(t) = e^{\bar{A}(t-t_o)}[Z(t_o) - Z_{ss}]Z^{-1}(t_o)x_o .$$

Equation (2.3.18) suggests a very simple propagation for computing x(t) at equal intervals of time, $\Delta t = (t_f-t_o)/N$, where N is the number of time-steps. The state vector at the next time-step is given by

$$x(t+\Delta t) = Z_{ss}a(t+\Delta t) + b(t+\Delta t) , \tag{2.3.19}$$

where a(t) and b(t) are propagated by the following linear difference equations:

$$a(t+\Delta t) = e^{-\bar{A}^T \Delta t} a(t) \ , \tag{2.3.20}$$

and

$$b(t+\Delta t) = e^{\bar{A}\Delta t} b(t) \ . \tag{2.3.21}$$

Computation of x(t) via (2.3.19), (2.3.20), and (2.3.21) requires roughly $3n^2$ flops per time step.

2.3.1.4 Control Trajectory Solution

The solution for the optimal control, u(t), can be written as

$$u(t) = - R^{-1} B^T P(t) x(t) \ ,$$

or

$$u(t) = - R^{-1} B^T [P_{ss} + Z^{-1}(t)] \Phi_2(t,t_o) x_o \ , \tag{2.3.22}$$

where the solutions for P(t) and x(t) have been introduced from (2.3.10) and (2.3.17), respectively. Simplification of (2.3.22), and substitution of Z(t) from (2.3.10) leads to

$$u(t) = - D_1 a(t) - D_2 b(t) \ , \tag{2.3.23}$$

where

$$D_1 = R^{-1} B^T [P_{ss} Z_{ss} + I] \ ,$$

$$D_2 = R^{-1} B^T P_{ss} \ ,$$

and a(t) and b(t) are defined in (2.3.18). Assuming that a(t) and b(t) have been computed for the state trajectory solution, the operation indicated by (2.3.23) requires 2pn flops.

2.3.1.5 Residual State Trajectory Solution

The response of the residual modes is governed by the differential equation

$$\dot{x}_r(t) = A_r x_r(t) + B_r u(t) \qquad (2.3.25)$$

$$= A_r x_r(t) - B_r D_1 a(t) - B_r D_2 b(t) ,$$

where A_r is the residual state dynamics matrix, B_r is the residual control influence matrix, and the solution for u(t) has been substituted from (2.3.23). The solution for $x_r(t)$ is then given by

$$x_r(t) = e^{A_r(t-t_o)} x_r(t_o) - G(A_r, B_r D_1, -\bar{A}^T, t, t_o)a_o - G(A_r, B_r D_2, \bar{A}, t, t_o)b_o \qquad (2.3.26)$$

where

$$G(A,B,C,t_2,t_1) \triangleq \int_{t_1}^{t_2} e^{A(t_2-\tau)} B e^{C(\tau-t_1)} d\tau .$$

It can be shown that the solution for $x_r(t)$ can be propagated at intervals of Δt by the following difference equation:

$$x_r(t+\Delta t) = e^{A_r \Delta t} x_r(t) - D_3 a(t) - D_4 b(t) , \qquad (2.3.27)$$

where

$$D_3 = G(A_r, B_r D_1, -\bar{A}^T, \Delta t, 0)$$

and

$$D_4 = G(A_r, B_r D_2, \bar{A}, \Delta t, 0) \ .$$

(See Appendix B for the calculation of D_3 and D_4).

Assuming that a(t) and b(t) are available from the state trajectory calculation, the above equation requires $n_r^2 + 2nn_r$ flops per time-step, where n_r is the number of residual states.

When the state vector x(t) is augmented to include the control, as in control-smoothing formulations (Appendix C), the above equations for the residual state response must be modified. The residual state differential equation becomes

$$\dot{x}_r(t) = A_r x_r(t) + B_r H_u x(t)$$

$$= A_r x_r(t) + B_r H_u Z_{ss} a(t) + B_r H_u b(t) \qquad (2.3.28)$$

where H_u is a selection matrix which selects the control vector partition of the augmented state. The solution for $x_r(t)$ can be shown to be

$$x_r(t) = e^{A_r(t-t_o)} x_r(t_o) + G(A_r, B_r H_u Z_{ss}, -\bar{A}^T, t, t_o) a_o$$

$$+ G(A_r, B_r H_u, \bar{A}, t, t_o) b_o \ , \qquad (2.3.29)$$

and the propagation equation becomes

$$x_r(t+\Delta t) = e^{A_r \Delta t} x_r(t) + D_3' a(t) + D_4' b(t) \ , \qquad (2.3.30)$$

where

$$D_3' = G(A_r, B_r H_u Z_{ss}, -\bar{A}^T, \Delta t, 0) \ ,$$

and

$$D_4' = G(A_r, B_r H_u, \bar{A}, \Delta t, 0) .$$

The computational work required to evaluate (2.3.30) is also $n_r^2 + 2nn_r$ flops per time-step.

The total amount of work required to propagate the states, controls, and residual states using the solutions based on Z(t) is $(3n^2) + (2pn) + (n_r^2 + 2nn_r)$.

2.3.1.6 Comparison with Alternative Closed-Form Solution

An alternative way of computing the system response under the assumptions of perfect plant knowledge and state estimation is to use the open-loop solution involving the state transition matrix. The total system state differential equation may be written as:

$$\begin{Bmatrix} \dot{x}_r(t) \\ \dot{x}(t) \\ \dot{\lambda}(t) \end{Bmatrix} = \begin{bmatrix} A_r & 0 & -E_r \\ 0 & A & -E \\ 0 & -Q & -A^T \end{bmatrix} \begin{Bmatrix} x_r(t) \\ x(t) \\ \lambda(t) \end{Bmatrix} , \qquad (2.3.31)$$

where $E_r = B_r R^{-1} B^T$. Upon forming the exponential of the matrix coefficient in (2.3.31), the system state may be propagated forward in time at intervals of Δt by the difference equation

$$\begin{Bmatrix} x_r(t+\Delta t) \\ x(t+\Delta t) \\ \lambda(t+\Delta t) \end{Bmatrix} = \begin{bmatrix} e^{A_r \Delta t} & \Gamma_1 & \Gamma_2 \\ 0 & \Theta_{11} & \Theta_{12} \\ 0 & \Theta_{21} & \Theta_{22} \end{bmatrix} \begin{Bmatrix} x_r(t) \\ x(t) \\ \lambda(t) \end{Bmatrix} , \qquad (2.3.32)$$

where $e^{A_r \Delta t}$, Γ_i, and Θ_{ij} are partitions of the exponential matrix

$$\begin{bmatrix} e^{A_r \Delta t} & \Gamma_1 & \Gamma_2 \\ 0 & \Theta_{11} & \Theta_{12} \\ 0 & \Theta_{21} & \Theta_{22} \end{bmatrix} = \exp\left\{ \begin{bmatrix} A_r & 0 & -E_r \\ 0 & A & -E \\ 0 & -Q & -A^T \end{bmatrix} \Delta t \right\} .$$

The initial condition for $\lambda(t)$ is obtained by introducing the terminal condition, $\lambda(t_f) = P_f x(t_f)$, into the lower two partitions of (2.3.32), with $t = t_o$ and $\Delta t = t_f - t_o$, and solving for $\lambda(t_o)$ to obtain:

$$\lambda(t_o) = [P_f\Theta_{12} - \Theta_{22}]^{-1}[\Theta_{21} - P_f\Theta_{11}]x_o . \qquad (2.3.33)$$

The control trajectory is then given directly by

$$u(t) = - R^{-1}B^T\lambda(t) . \qquad (2.3.34)$$

The amount of work involved in evaluating (2.3.32) and (2.3.34) is roughly $4n^2 + pn + n_r^2 + 2nn_r$ per time-step, where the sparsity of the matrix exponential in (2.3.32) has been taken into account. For control problems where the plant state is augmented by the control vector, such as in control-smoothing formulations, the matrix coefficient in (2.3.31) is altered so that the equation becomes

$$\begin{Bmatrix} \dot{x}_r(t) \\ \dot{x}(t) \\ \dot{\lambda}(t) \end{Bmatrix} = \begin{bmatrix} A_r & B_rH_u & 0 \\ 0 & A & -E \\ 0 & -Q & -A^T \end{bmatrix} \begin{Bmatrix} x_r(t) \\ x(t) \\ \lambda(t) \end{Bmatrix} . \qquad (2.3.35)$$

Compared with the solution derived from the Z(t) equation, this method requires more work per time-step if $p < n$ or the same work per time step if $p = n$. In comparing the initial calculations, the exponential matrix of (2.3.32) is larger than those required in the Z(t) type solutions. Since the amount of work required to exponentiate a matrix is on the order of the third power of the matrix dimension, and also depends on the infinity norm of the matrix, it is advantageous to exponentiate several smaller-sized matrices rather than exponentiate one large matrix.

Thus, the closed-form solutions based on Z(t) are, on the whole, more efficient than the open-loop solutions.

2.3.2 Linear Optimal Controller with Terminal Constraints

2.3.2.1 Optimal Control Problem and Necessary Conditions

The structure of the type of feedback control system discussed in this section is shown in Figure 2-2. Mathematically, the optimal control problem is defined as the minimization of the performance index

$$J = \frac{1}{2}\|y(t_f)\|^2_{P'} + \frac{1}{2}\int_{t_o}^{t_f} [\|y(t)\|^2_{Q'} + \|u(t)\|^2_R]dt \ , \qquad (2.3.36)$$

subject to

$$\psi(t_f) = My(t_f) - \psi_d = 0 \ , \qquad (2.3.37)$$

$$\dot{x}(t) = Ax(t) + Bu(t) \ , \quad x(t_o) = x_o \ , \qquad (2.3.38)$$

and

$$y(t) = Cx(t) \ , \qquad (2.3.39)$$

where $\psi(t_f)$ is the q-dimensional terminal constraint function, M is the terminal constraint matrix which selects linear combinations of the output states to be constrained at the final time, ψ_d is a q-dimensional vector of desired terminal conditions, and all other variables and matrices are defined in Section 2.3.1.1.

The necessary conditions for the minimization of (2.3.36) can be shown to be [14]

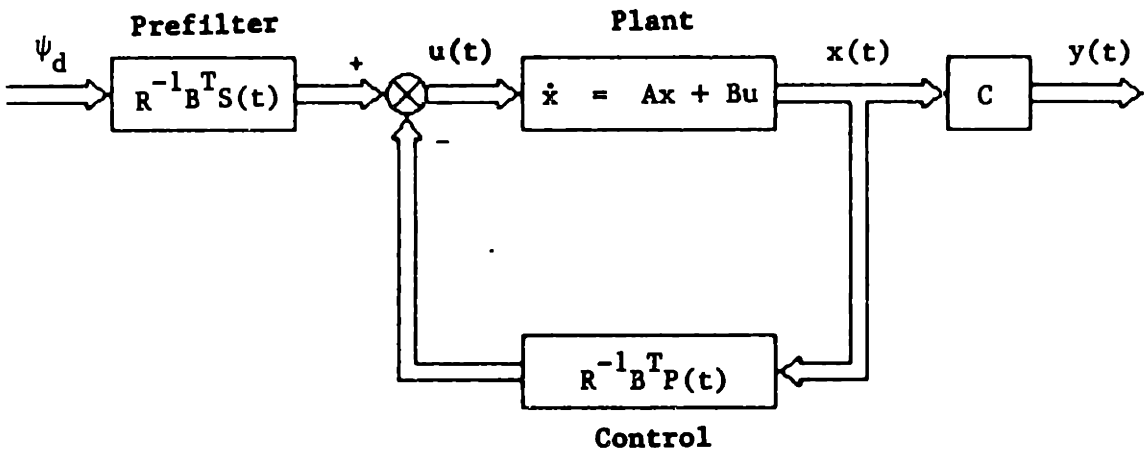


Figure 2-2. Block Diagram for the Linear Optimal Controller with Terminal Constraints

$$\dot{x}(t) = Ax(t) + Bu(t) , \qquad x(t_o) = x_o , \tag{2.3.40}$$

$$\dot{\lambda}(t) = - Qx(t) - A^T\lambda(t) , \quad \lambda(t_f) = C^TM^T\nu + P_fx(t_f) , \tag{2.3.41}$$

$$u(t) = - R^{-1}B^T\lambda(t) , \tag{2.3.42}$$

and

$$\psi(t_f) = MCx(t_f) - \psi_d = 0 , \tag{2.3.43}$$

where ν is the q-dimensional constant Lagrange multiplier vector for the terminal constraint. The terminal conditions for the costate and the constraint function may be written in matrix-vector form as

$$\begin{Bmatrix} \lambda(t_f) \\ \psi(t_f)=0 \end{Bmatrix} = \begin{bmatrix} \bar{P} & \bar{S} & 0 \\ \bar{S}^T & \bar{G} & I \end{bmatrix} \begin{Bmatrix} x(t_f) \\ \nu \\ -\psi_d \end{Bmatrix} , \tag{2.3.44}$$

where

$$\bar{P} = P_f , \quad \bar{S} = C^TM^T , \quad \text{and } \bar{G} = 0 .$$

2.3.2.2 Feedback Solution

In order to obtain a feedback solution for the control problem, it is convenient to express $\lambda(t)$ in terms of $x(t)$ and ψ_d. To this end, the terminal conditions of (2.3.39) are manipulated into the form

$$\begin{Bmatrix} \lambda(t_f) \\ \nu \end{Bmatrix} = \begin{bmatrix} \bar{P} - \bar{S}\bar{G}^{-1}\bar{S}^T & - \bar{S}\bar{G}^{-1} \\ - \bar{G}^{-1}\bar{S}^T & - \bar{G}^{-1} \end{bmatrix} \begin{Bmatrix} x(t_f) \\ - \psi_d \end{Bmatrix} . \tag{2.3.45}$$

The feedback control is then obtained by assuming a solution for $\lambda(t)$ and ν of the form

$$\begin{Bmatrix} \lambda(t) \\ \nu \end{Bmatrix} = \begin{bmatrix} P(t) & S(t) \\ S^T(t) & G(t) \end{bmatrix} \begin{Bmatrix} x(t) \\ -\psi_d \end{Bmatrix} , \tag{2.3.46}$$

where

$$P(t_f) = \bar{P} - \bar{\bar{S}}\bar{\bar{G}}^{-1}\bar{\bar{S}}^T ,$$

$$S(t_f) = -\bar{\bar{S}}\bar{\bar{G}}^{-1} ,$$

and

$$G(t_f) = -\bar{\bar{G}}^{-1} .$$

Note that since $\bar{G} = 0$, the matrices P(t), S(t), and G(t) are infinitely large at the final time, or treated as nonexistent. However, this is entirely consistent with the problem formulation, which requires the terminal constraint to be met with zero error.

The differential equations for P(t) and S(t) are obtained by substituting the expression for $\lambda(t)$ from (2.3.46) into (2.3.40) through (2.3.42), and grouping coefficients of x(t) and ψ_d to obtain:

$$\dot{P}(t) = -P(t)A - A^T P(t) + P(t)EP(t) - Q , \tag{2.3.47}$$

and

$$\dot{S}(t) = -[A - EP(t)]^T S(t) , \tag{2.3.48}$$

where

$$E = BR^{-1}B^T .$$

Equation (2.3.47) is the Type 1 differential matrix Riccati equation, and (2.3.48) is a Type 2 differential equation of Section 2.2.2. The differential equation for G(t) is obtained by differentiating the

expression for ν in (2.3.46), treating ν and ψ_d as constants. This yields

$$\dot{G}(t) = S^T(t)ES(t) . \tag{2.3.49}$$

As shown in the solution of the optimal regulator problem, and in Section 2.2.1, the solution for P(t) is

$$P(t) = P_{ss} + Z^{-1}(t) , \tag{2.3.50}$$

where

$$Z(t) = Z_{ss} + e^{\bar{A}(t-t_o)}[Z(t_o) - Z_{ss}]e^{\bar{A}^T(t-t_o)} ,$$

$$Z(t_o) = Z_{ss} + e^{-\bar{A}(t_f-t_o)}[Z(t_f) - Z_{ss}]e^{-\bar{A}^T(t_f-t_o)} ,$$

and

$$Z(t_f) = [P(t_f) - P_{ss}]^{-1}.$$

The solution for S(t) is obtained by using Tables 2-1 and 2-2 for Type 2 differential equations, leading to:

$$S(t) = \Phi_1(t,t_o)S(t_o)$$

$$= Z^{-1}(t)e^{\bar{A}(t-t_o)}Z(t_o)S(t_o) . \tag{2.3.51}$$

Similarly, the solution for G(t) is found to be

$$G(t) = S^T(t)Z(t)S(t) + G(t_o) - S^T(t_o)Z(t_o)S(t_o) , \tag{2.3.52}$$

from Table 2-1 for Type 4 differential equations.

The initial conditions for P(t), S(t) and G(t) are obtained by transforming the terminal conditions, shown after (2.3.46), into initial

conditions via the solutions of P(t), S(t), and G(t), and applying the matrix inversion lemma to eliminate terms involving $\bar{G}^{-1}$. The detailed steps are shown in Appendix D and summarized as follows:

$$Z(t_f) = \tilde{P}^{-1} - \tilde{P}^{-1}\bar{S}[\bar{S}^T\tilde{P}^{-1}\bar{S}-\bar{G}]^{-1}\bar{S}^T\tilde{P}^{-1} \quad , \tag{2.3.53}$$

$$S(t_o) = Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}\tilde{P}^{-1}\bar{S}[\bar{S}^T\tilde{P}^{-1}\bar{S}-\bar{G}]^{-1} \quad , \tag{2.3.54}$$

$$G(t_o) = S^T(t_o)Z(t_o)S(t_o) + [\bar{S}^T\tilde{P}^{-1}\bar{S}-\bar{G}]^{-1} \quad , \tag{2.3.55}$$

where

$$\tilde{P} = P_f - P_{ss} \quad ,$$

and $Z(t_o)$ is computed using the equation following (2.3.50). For numerical stability during simulation and application of the control system, it is advantageous to set $\bar{G}$ to a small non-zero matrix, i.e. $\bar{G} = -\varepsilon I$, where ε is a small positive number.

2.3.2.3 Propagation of the Feedback Gains

From (2.3.42) and (2.3.46), the feedback form for the optimal control is given by

$$u(t) = -R^{-1}B^TP(t)x(t) + R^{-1}B^TS(t)\psi_d \quad . \tag{2.3.56}$$

Thus, the matrices P(t) and S(t) are needed for the feedback control, while the solution for G(t) is not used. The propagation of the Riccati solution, P(t), is shown in Section 2.3.1.2. An efficient propagation method for S(t) is given by

$$S(t+\Delta t) = Z^{-1}(t+\Delta t)\tilde{S}(t+\Delta t) , \tag{2.3.57}$$

and

$$\tilde{S}(t+\Delta t) = e^{\bar{A}\Delta t}\tilde{S}(t) , \tag{2.3.58}$$

where

$$\tilde{S}(t_o) = Z(t_o)S(t_o) .$$

Equations (2.3.57) and (2.3.58) require qn^2 flops each, assuming the Cholesky decomposition of $Z(t+\Delta t)$ has already been computed during the propagation for $P(t)$. Thus, the total amount of computation needed for the propagation of $S(t)$ is about $2qn^2$ flops.

For comparison, a Kalman-Englar type of propagation is presented as follows. The state-costate system can be propagated via

$$\begin{Bmatrix} x(t+\Delta t) \\ \lambda(t+\Delta t) \end{Bmatrix} = \begin{bmatrix} \theta_{11} & \theta_{12} \\ \theta_{12} & \theta_{22} \end{bmatrix} \begin{Bmatrix} x(t) \\ \lambda(t) \end{Bmatrix} , \tag{2.3.59}$$

where $\theta_{ij} = \theta_{ij}(t+\Delta t,t)$ are partitions of the transition matrix of (2.3.13). On eliminating $x(t)$ from (2.3.59) and replacing $\lambda(t)$ by $P(t)x(t) - S(t)\psi_d$ from (2.3.46), one obtains

$$\lambda(t+\Delta t) = P(t+\Delta t)x(t+\Delta t) - [\theta_{22} - P(t+\Delta t)\theta_{12}]S(t)\psi_d , \tag{2.3.60}$$

where

$$P(t+\Delta t) = [\theta_{21} + \theta_{22}P(t)][\theta_{11} + \theta_{12}P(t)]^{-1} .$$

Comparison of (2.3.60) with the upper part of (2.3.46) leads to a Kalman-Englar type of propagation for $S(t)$:

$$S(t+\Delta t) = [\theta_{22} - P(t+\Delta t)\theta_{12}]S(t) \quad . \tag{2.3.61}$$

Assuming that $P(t+\Delta t)$ is already computed, (2.3.61) requires either $n^3 + qn^2$ or $3qn^2$ flops, depending on the order of multiplication. The first operation count corresponds to performing the product and subtraction within the brackets before multiplication with S(t). The second operation count corresponds to the following form of (2.3.61):

$$S(t+\Delta t) = \theta_{22}S(t) - P(t+\Delta t)[\theta_{12}S(t)] \quad . \tag{2.3.62}$$

where the product within the brackets is computed first.

For $q < n/2$, (2.3.62) is more efficient than (2.3.61); for $q > n/2$, (2.3.61) is more efficient than (2.3.62). However, for all values of $q < n$, the propagation for the Z(t) type solution of (2.3.57) and (2.3.58) is most efficient. For $q < n/2$, propagation via (2.3.57) and (2.3.58) is 30% faster than using (2.3.62). Comparing (2.3.61) with (2.3.57) and (2.3.58) , the advantage becomes less as q increases above n/2, until $q = n$, when the operation counts become the same. Thus, the closed-form solution for S(t) involving Z(t) is more efficiently propagated than a Kalman-Englar type of solution.

The propagation equations for P(t) and S(t) are needed only for the implementation of the feedback control system, or for parametric studies. For simulation of the nominal system response, P(t) and S(t) need not be propagated.

2.3.2.4 State Trajectory Solution

With the assumptions of perfect plant knowledge and perfect state estimation, the state response is governed by

$$\dot{x}(t) = Ax(t) - E\lambda(t) \tag{2.3.63}$$

$$= [A - EP(t)]x(t) + ES(t)\psi_d ,$$

which is obtained by introducing $\lambda(t)$ from (2.3.46) into (2.3.42) and (2.3.40). The above equation is classified as Type 3 of Section 2.2.3. From Table 2-1, the solution for x(t) is

$$x(t) = \Phi_2(t,t_o)x_o + \int_{t_o}^{t} \Phi_2(t,\tau)ES(\tau)d\tau\psi_d . \tag{2.3.64}$$

Substitution of (2.3.51) into the above equation leads to

$$x(t) = \Phi_2(t,t_o)x_o + \int_{t_o}^{t} \Phi_2(t,\tau)E\Phi_1(\tau,t_o)d\tau S(t_o)\psi_d . \tag{2.3.65}$$

The integral term of (2.3.65) may be simplified as follows. Using the expressions for $\Phi_1(t,t_o)$ and $\Phi_2(t,t_o)$ from Table 2-2, one can write

$$\int_{t_o}^{t} \Phi_2(t,\tau)E\Phi_1(\tau,t_o)d\tau$$

$$= Z(t)e^{-\bar{A}^T(t-t_o)} \int_{t_o}^{t} e^{\bar{A}^T(\tau-t_o)} Z^{-1}(\tau)EZ^{-1}(\tau)e^{\bar{A}(\tau-t_o)} d\tau Z(t_o) . \tag{2.3.66}$$

Defining the expression

$$Y(t) = \int_{t_o}^{t} e^{\bar{A}^T(\tau-t_o)} Z^{-1}(\tau)EZ^{-1}(\tau)e^{\bar{A}(\tau-t_o)} d\tau , \tag{2.3.67}$$

one can write

$$\dot{Y}(t) = e^{\bar{A}^T(t-t_o)} Z^{-1}(t) E Z^{-1}(t) e^{\bar{A}(t-t_o)} , \qquad (2.3.68)$$

which has the solution

$$Y(t) = e^{\bar{A}^T(t-t_o)} Z^{-1}(t) e^{\bar{A}(t-t_o)} - Z^{-1}(t_o) , \qquad (2.3.69)$$

which is very similar in form to the Type 5 equations of Section 2.2.5. Substitution of (2.3.69) into (2.3.66) leads to the identity

$$\int_{t_o}^{t} \Phi_2(t,\tau) E \Phi_1(\tau,t_o) d\tau \equiv [e^{\bar{A}(t-t_o)} - \Phi_2(t,t_o)] Z(t_o) . \qquad (2.3.70)$$

Introducing (2.3.70) into (2.3.65) leads to

$$x(t) = \Phi_2(t,t_o)[x_o - Z(t_o)S(t_o)\psi_d] + e^{\bar{A}(t-t_o)} Z(t_o)S(t_o)\psi_d . \qquad (2.3.71)$$

Upon substitution of the expressions for $\Phi_2(t,t_o)$ and $Z(t)$ into (2.3.71), and simplification, the solution for $x(t)$ is finally reduced to

$$x(t) = Z_{ss} a(t) + b(t) , \qquad (2.3.72)$$

where

$$a(t) = e^{-\bar{A}^T(t-t_o)} [Z^{-1}(t_o)x_o - S(t_o)\psi_d] ,$$

and

$$b(t) = e^{\bar{A}(t-t_o)} [x_o - Z_{ss}(Z^{-1}(t_o)x_o - S(t_o)\psi_d)] .$$

To compute $x(t)$ at equal intervals of time, $\Delta t = (t_f - t_o)/N$, where N is the number of time-steps, one can use the following equations:

$$x(t+\Delta t) = Z_{ss}a(t+\Delta t) + b(t+\Delta t) , \tag{2.3.73}$$

$$a(t+\Delta t) = e^{-\bar{A}^T\Delta t}a(t) , \tag{2.3.74}$$

and

$$b(t+\Delta t) = e^{\bar{A}\Delta t}b(t) . \tag{2.3.75}$$

Computation of x(t) via (2.3.73) through (2.3.75) requires $3n^2$ flops.

2.3.2.5 Control Trajectory Solution

The solution for the optimal control trajectory is given in (2.3.56) by

$$u(t) = - R^{-1}B^T[P(t)x(t) - S(t)\psi_d] . \tag{2.3.76}$$

Substituting the solutions for P(t) from (2.3.50), x(t) from (2.3.72), and S(t) from (2.3.51) into (2.3.76) and simplifying yields

$$u(t) = - D_1a(t) - D_2b(t) , \tag{2.3.77}$$

where

$$D_1 = R^{-1}B^T[P_{ss}Z_{ss} + I] ,$$

and

$$D_2 = R^{-1}B^TP_{ss} .$$

The operation indicated by (2.3.77) requires about 2pn flops per time-step.

2.3.2.6 Residual State Trajectory Solution

The residual state differential equation is given by

$$\dot{x}_r(t) = A_r x_r(t) + B_r u(t)$$

$$= A_r x_r(t) - B_r D_1 a(t) - B_r D_2 b(t) \ . \qquad (2.3.78)$$

Comparison with (2.3.25) shows that the above equation has the same form as in the optimal linear regulator, except that the initial conditions on a(t) and b(t) are different. The solution for $x_r(t)$ is then given by (2.3.26), and the propagation equation given by (2.3.27). Similarly, for control-smoothing formulations, the residual state solution and propagation are given by (2.3.29) and (2.3.30), respectively. The computational work required for the residual mode response is $n_r^2 + 2nn_r$ flops per time-step, as in the linear optimal regulator. The total amount of work required for the state, control, and residual propagation is thus $(3n^2) + (2pn) + (n_r^2 + 2nn_r)$ flops per time-step.

2.3.2.7 Comparison with Alternative Closed-Form Solution

An alternative closed-form solution for the state, control, and residual state trajectories when the plant and states are perfectly known is the state transition matrix method, which corresponds to the open-loop solution of the terminal controller. Since the differential equations for the state, costate, and residual state are identical to those derived for the linear optimal regulator, the propagation equations for the terminal controller are also given by (2.3.32) and (2.3.34). However, the terminal boundary conditions are not the same. To obtain the initial condition for $\lambda(t)$, one first writes the following relationship for the initial and final states and costates:

$$\begin{Bmatrix} x(t_f) \\ \lambda(t_f) \end{Bmatrix} = \begin{bmatrix} \Theta_{11} & \Theta_{12} \\ \Theta_{21} & \Theta_{22} \end{bmatrix} \begin{Bmatrix} x(t_o) \\ \lambda(t_o) \end{Bmatrix}, \tag{2.3.79}$$

where $\Theta_{ij} = \Theta_{ij}(t_f, t_o)$ are partitions of the final time state transition matrix. On introducing the terminal condition for $\lambda(t)$ from (2.3.41), and including the terminal constraint condition, (2.3.79) becomes

$$\begin{bmatrix} I & 0 \\ P_f & C^T M^T \\ MC & 0 \end{bmatrix} \begin{Bmatrix} x(t_f) \\ \nu \end{Bmatrix} = \begin{bmatrix} \Theta_{11} & \Theta_{12} & 0 \\ \Theta_{21} & \Theta_{22} & 0 \\ 0 & 0 & I \end{bmatrix} \begin{Bmatrix} x_o \\ \lambda(t_o) \\ \psi_d \end{Bmatrix}. \tag{2.3.80}$$

Re-arranging the above equation, one obtains the expression for the unknown variables:

$$\begin{Bmatrix} x(t_f) \\ \lambda(t_o) \\ \nu \end{Bmatrix} = \begin{bmatrix} I & -\Theta_{12} & 0 \\ P_f & -\Theta_{22} & C^T M^T \\ MC & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} \Theta_{11} & 0 \\ \Theta_{21} & 0 \\ 0 & I \end{bmatrix} \begin{Bmatrix} x_o \\ \psi_d \end{Bmatrix}. \tag{2.3.81}$$

The solution for $\lambda(t_o)$ can then be used along with x_o, and $x_r(t_o)$ as initial states for the propagation equation, (2.3.32).

The comparison of the amount of work required to propagate the states, controls, and residual states for both the solutions involving Z(t) and the open-loop solution is the same as in the linear optimal regulator, namely $3n^2 + 2pn + n_r^2 + 2nn_r$ for the solutions involving Z(t) and $4n^2 + pn + n_r^2 + 2nn_r$ for the open-loop solution.

Numerical results for this type of control formulation are shown in Ref. [47-49].

2.3.3 Tracking/Disturbance Accommodating Controller

2.3.3.1 Optimal Control Problem and Necessary Conditions

The structure of the type of controller discussed in this section is shown in Figure 2-3. Mathematically, the optimal control problem is defined as the minimization of the performance index

$$J = \frac{1}{2}\|y^*(t_f)-y(t_f)\|^2_{P'} + \frac{1}{2}\int_{t_o}^{t_f}[\|y^*(t)-y(t)\|^2_{Q'} + \|u(t)\|^2_R]dt \ , \tag{2.3.82}$$

subject to

$$\dot{x}(t) = Ax(t) + Bu(t) + \Lambda\beta(t) \ , \quad x(t_o) = x_o \ , \tag{2.3.83}$$

and

$$y(t) = Cx(t) \ , \tag{2.3.84}$$

where $y^*(t)$ is the m-dimensional reference output state, Λ is the $(n \times n_d)$ disturbance influence matrix, β is the n_d-dimensional disturbance state, and the other variables and constants are as defined in Section 2.3.1.1.

Let us assume that the disturbance state is governed by the following dynamics:

$$\dot{\beta}(t) = L\beta(t) \ , \quad \beta(t_o) = \beta_o \ , \tag{2.3.85}$$

so that the instantaneous disturbance state may be written as

$$\beta(t) = e^{L(t-t_o)}\beta_o \ . \tag{2.3.86}$$

Equation (2.3.85) represents the dynamics of a linear time-invariant system. In many cases, a known disturbance may be modelled as Fourier

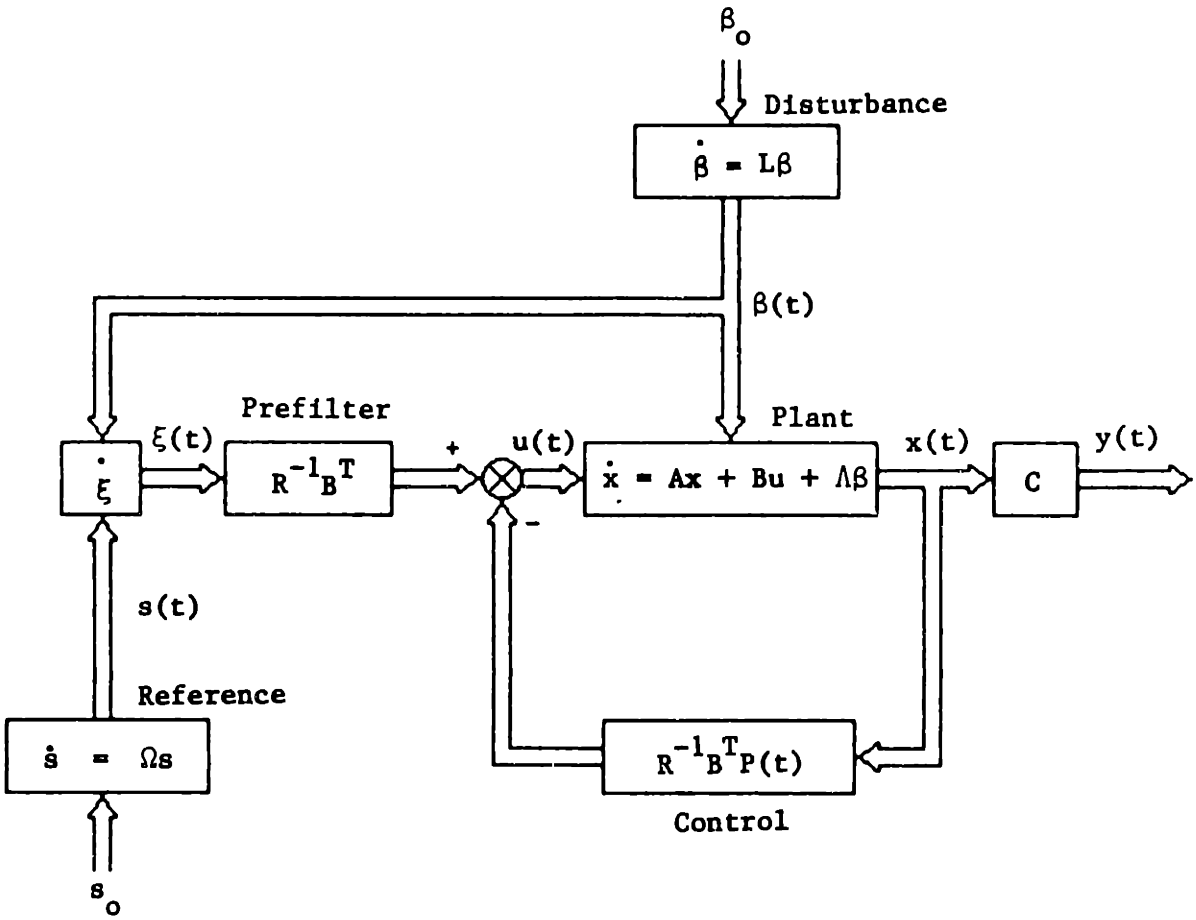


Figure 2-3. Block Diagram for the Tracking/Disturbance Accommodating Controller

series. In such a case, the rows of Λ consist of the Fourier series coefficients for the individual disturbances, and L is a constant block diagonal matrix:

$$L = \text{Block diag} \left[0 , \begin{bmatrix} 0 & 1 \\ -\omega_1^2 & 0 \end{bmatrix} , \cdots , \begin{bmatrix} 0 & 1 \\ -\omega_k^2 & 0 \end{bmatrix} \right] , \tag{2.3.87}$$

where $n_d = 2k + 1$.

The reference output state may be represented as the output of a dynamic system:

$$y^*(t) = Fs(t) , \tag{2.3.88}$$

$$\dot{s}(t) = \Omega s(t) , \tag{2.3.89}$$

and s(t) has the solution

$$s(t) = e^{\Omega(t-t_o)} s_o . \tag{2.3.90}$$

Since we are mainly interested in finite-time problems, (2.3.89) would represent the dynamics of a state-costate system corresponding to the open-loop solution of an optimal control problem. For this case, the matrix F can be replaced by the product $C^* H_x$, where $H_x = [I \ 0]$ is a selection matrix for the state vector portion of s(t), and C^* is an output matrix. The open-loop dynamics matrix Ω then has the form

$$\Omega = \begin{bmatrix} A^* & -B^*(R^*)^{-1}(B^*)^T \\ -Q^* & -(A^*)^T \end{bmatrix} \tag{2.3.91}$$

where A^*, B^*, Q^*, and R^* represent the state dynamics, control influence, state weight, and control weight matrices for the reference state.

Denoting the dimension of $s(t)$ by $2n_o$, one may have $n_o < n$, which means that the higher-dimensioned state $x(t)$ is controlled to approximate the dynamics of a lower-dimensioned system. However, if control smoothing is used, it is desirable to have the same degree of control smoothing in both the tracking/disturbance accommodating controller and the nominal state dynamics.

The necessary conditions for the tracking/disturbance accommodating control problem can be shown to be [14]:

$$\dot{x}(t) = Ax(t) + Bu(t) + \Lambda\beta(t) , \qquad x(t_o) = x_o , \tag{2.3.92}$$

$$\dot{\lambda}(t) = - Qx(t) + \tilde{Q}x^*(t) - A^T\lambda(t) , \tag{2.3.93}$$

$$\lambda(t_f) = P_f x(t_f) - \tilde{P}_f x^*(t_f) , \tag{2.3.94}$$

$$u(t) = - R^{-1}B^T\lambda(t) , \tag{2.3.95}$$

where

$$\tilde{Q} = C^T Q' C^* ,$$

$$\tilde{P}_f = C^T P' C^* ,$$

and

$$x^*(t) = H_x s(t) .$$

2.3.3.2 Solution for the Feedback Control

The closed-loop control is determined by assuming that

$$\lambda(t) = P(t)x(t) - \xi(t) , \tag{2.3.96}$$

where $P(t)$ is an $(n \times n)$ matrix and $\xi(t)$ is an n-dimensional vector, both of which are to be determined. The optimal control function of (2.3.95) can then be expressed as:

$$u(t) = - R^{-1}B^T[P(t)x(t) - \xi(t)] . \tag{2.3.97}$$

Substitution of (2.3.96) into (2.3.92) through (2.3.95) leads to the differential equations and boundary conditions for $P(t)$ and $\xi(t)$:

$$\dot{P}(t) = - P(t)A - A^TP(t) + P(t)EP(t) - Q , \quad P(t_f) = P_f , \tag{2.3.98}$$

and

$$\dot{\xi}(t) = - [A - EP(t)]^T\xi(t) - \tilde{Q}x^*(t) + P(t)A\beta(t) , \quad \xi(t_f) = \tilde{P}_f x^*(t_f) . \tag{2.3.99}$$

Since the differential equation and boundary condition for $P(t)$ are identical to those for the linear optimal regulator ((2.3.9)), the solutions are the same, and are given by (2.3.10). The differential equation for $\xi(t)$ is identified as a nonhomogeneous Type 2 differential equation of Section 2.2.2. Referring to Table 2-1, the solution for $\xi(t)$ is

$$\xi(t) = \Phi_1(t,t_o)\xi(t_o) - I_1(t) + I_2(t) , \tag{2.3.100}$$

where

$$I_1(t) = \int_{t_o}^{t} \Phi_1(t,\tau)\tilde{Q}x^*(\tau)d\tau ,$$

and

$$I_2(t) = \int_{t_o}^{t} \Phi_1(t,\tau)P(\tau)\Lambda\beta(\tau)d\tau \ .$$

Using the definition for $\Phi_1(t,t_o)$ from Table 2-2, and the expression for $x^*(t)$, one can write the first integral of (2.3.100) as

$$I_1(t) = Z^{-1}(t) \int_{t_o}^{t} e^{\bar{A}(t-\tau)} Z(\tau)\tilde{Q}H_x e^{\Omega(\tau-t_o)} d\tau s_o \ . \qquad (2.3.101)$$

Substituting the initial-time form of the solution for Z(t) from (2.3.10) into the integral part of (2.3.101) and simplifying leads to

$$I_1(t) = Z^{-1}(t)G(\bar{A},K_1,\Omega,t,t_o)s_o$$

$$+ Z^{-1}(t)[Z(t) - Z_{ss}]G(-\bar{A}^T,K_2,\Omega,t,t_o)s_o \ , \qquad (2.3.102)$$

where

$$K_1 = Z_{ss}\tilde{Q}H_x \ ,$$

$$K_2 = \tilde{Q}H_x \ ,$$

and $G(A,B,C,t_2,t_1)$ is defined following (2.3.26). Similarly, the second integral of (2.3.100) may be written as

$$I_2(t) = Z^{-1}(t) \int_{t_o}^{t} e^{\bar{A}(t-\tau)} Z(\tau)[P_{ss} + Z^{-1}(\tau)]\Lambda e^{L(\tau-t_o)} d\tau\beta_o \ . \qquad (2.3.103)$$

Substitution of the initial-time form of the solution for Z(t) from (2.3.10) into (2.3.103), and simplifying, leads to

$$I_2(t) = Z^{-1}(t)G(\bar{A},K_3,L,t,t_o)\beta_o$$

$$+ Z^{-1}(t)[Z(t) - Z_{ss}]G(-\bar{A}^T,K_4,L,t,t_o)\beta_o \ , \qquad (2.3.104)$$

where

$$K_3 = [Z_{ss}P_{ss} + I]\Lambda \ ,$$

and

$$K_4 = P_{ss}\Lambda \ .$$

The initial condition for $\xi(t)$ is obtained by substituting the final condition from (2.3.99) into (2.3.100), giving

$$\bar{P}_f x^*(t_f) = \Phi_1(t_f,t_o)\xi(t_o) - I_1(t_f) + I_2(t_f) \ . \qquad (2.3.105)$$

Solving for $\xi(t_o)$ yields

$$\xi(t_o) = \Phi_1(t_o,t_f)[\bar{P}_f x^*(t_f) + I_1(t_f) - I_2(t_f)] \ . \qquad (2.3.106)$$

From (2.3.100), (2.3.102), and (2.3.104), the solution for $\xi(t)$ can be written as

$$\xi(t) = \Phi_1(t,t_o)\xi(t_o) + Z^{-1}(t)[G(\bar{A},K_3,L,t,t_o)\beta_o - G(\bar{A},K_1,\Omega,t,t_o)s_o]$$

$$+ Z^{-1}(t)[Z(t) - Z_{ss}][G(-\bar{A}^T,K_4,L,t,t_o)\beta_o - G(-\bar{A}^T,K_2,\Omega,t,t_o)s_o] \ . \qquad (2.3.107)$$

The propagation equations for $\xi(t)$ are shown in Section 2.3.3.7.

2.3.3.3 State Trajectory Solution

Under the assumptions of perfect plant knowledge and perfect state estimation, the modified equation for x(t) is obtained by introducing (2.3.97) into (2.3.92), giving

$$\dot{x}(t) = [A - EP(t)]x(t) + E\xi(t) + \Lambda\beta(t) . \qquad (2.3.108)$$

Recognizing that this is a nonhomogeneous differential equation of Type 3 of Section 2.2.3, the solution for x(t) is obtained from Table 2-1 as

$$x(t) = \Phi_2(t,t_o)x_o + I_3(t) + I_4(t) , \qquad (2.3.109)$$

where

$$I_3(t) = \int_{t_o}^{t} \Phi_2(t,\tau)E\xi(\tau)d\tau ,$$

and

$$I_4(t) = \int_{t_o}^{t} \Phi_2(t,\tau)\Lambda\beta(\tau)d\tau .$$

Substituting the solution for $\xi(t)$ from (2.3.100) into $I_3(t)$ above yields

$$I_3(t) = \int_{t_o}^{t} \Phi_2(t,\tau)E\Phi_1(\tau,t_o)d\tau\xi(t_o)$$

$$- \int_{t_o}^{t} \Phi_2(t,\tau)E \int_{t_o}^{\tau} \Phi_1(\tau,\varepsilon)\tilde{Q}x^*(\varepsilon)d\varepsilon\, d\tau$$

$$+ \int_{t_o}^{t} \Phi_2(t,\tau)E \int_{t_o}^{t} \Phi_1(\tau,\varepsilon)P(\varepsilon)\Lambda\beta(\varepsilon)d\varepsilon\, d\tau . \qquad (2.3.110)$$

The first integral term in the above expression for $I_3(t)$ may be simplified using (2.3.70). The remaining two double integral terms may be simplified as follows. Defining the function K(t) as

$$K(t) = \int_{t_o}^{t} \Phi_2(t,\tau) E \int_{t_o}^{t} \Phi_1(\tau,\varepsilon) f(\varepsilon) d\varepsilon \, d\tau \ , \qquad (2.3.111)$$

where f(t) is a general forcing term, one can substitute the expressions for $\Phi_1(t,t_o)$ and $\Phi_2(t,t_o)$ from Table 2-2 to obtain

$$K(t) = Z(t) e^{-\bar{A}^T t} Y(t) \ , \qquad (2.3.112)$$

where

$$Y(t) = \int_{t_o}^{t} e^{\bar{A}^T \tau} Z^{-1}(\tau) E Z^{-1}(\tau) e^{\bar{A}\tau} g(\tau) d\tau \ ,$$

and

$$g(t) = \int_{t_o}^{t} e^{-\bar{A}\tau} Z(\tau) f(\tau) d\tau \ .$$

The differential equation for Y(t) is then

$$\dot{Y}(t) = e^{\bar{A}^T t} Z^{-1}(t) E Z^{-1}(t) e^{\bar{A}t} g(t) \ , \quad Y(t_o) = 0 \ . \qquad (2.3.113)$$

Note that the above differential equation is similar to the Type 5 differential equation of Section 2.2.5. Assuming a solution for Y(t) of the form

$$Y(t) = e^{\bar{A}^T t} Z^{-1}(t) e^{\bar{A}t} g(t) + k(t) \ , \qquad (2.3.114)$$

where k(t) is an unknown function, one can differentiate (2.3.114) and, substituting into (2.3.113), obtain the following differential equation for k(t):

$$\dot{k}(t) = -e^{\bar{A}^T t} f(t) , \quad k(t_o) = 0 , \tag{2.3.115}$$

which has the formal solution

$$k(t) = -\int_{t_o}^{t} e^{\bar{A}^T \tau} f(\tau) d\tau . \tag{2.3.116}$$

Substituting (2.3.116) into (2.3.114), and the result into (2.3.112), one can write the following identity:

$$\int_{t_o}^{t} \Phi_2(t,\tau) E \int_{t_o}^{\tau} \Phi_1(\tau,\varepsilon) f(\varepsilon) d\varepsilon \, d\tau$$

$$\equiv \int_{t_o}^{t} e^{\bar{A}(t-\tau)} Z(\tau) f(\tau) d\tau - Z(t) \int_{t_o}^{t} e^{-\bar{A}^T(t-\tau)} f(\tau) d\tau . \tag{2.3.117}$$

Using (2.3.70) and (2.3.117), one can now simplify the expression for $I_3(t)$ in (2.3.110) to

$$I_3(t) = [e^{\bar{A}(t-t_o)} - \Phi_2(t,t_o)] Z(t_o) \xi(t_o)$$

$$- \int_{t_o}^{t} e^{\bar{A}(t-\tau)} Z(\tau) \bar{Q} H_x e^{\Omega(\tau-t_o)} d\tau s_o$$

$$+ Z(t) \int_{t_o}^{t} e^{-\bar{A}^T(t-\tau)} \bar{Q} H_x e^{\Omega(\tau-t_o)} d\tau s_o$$

$$+ \int_{t_o}^{t} e^{\bar{A}(t-\tau)} Z(\tau) [P_{ss} + Z^{-1}(\tau)] \Lambda e^{L(\tau-t_o)} d\tau \beta_o$$

$$- Z(t) \int_{t_o}^{t} e^{-\bar{A}^T(t-\tau)} [P_{ss} + Z^{-1}(\tau)] \Lambda e^{L(\tau-t_o)} d\tau \beta_o \ , \tag{2.3.118}$$

where $x^*(t)$ and $\beta(t)$ have been written in their exponential matrix forms. Similarly, from Table 2-2, $I_4(t)$ can be written as

$$I_4(t) = Z(t) \int_{t_o}^{t} e^{-\bar{A}^T(t-\tau)} Z^{-1}(\tau) \Lambda e^{L(\tau-t_o)} d\tau \beta_o \ . \tag{2.3.119}$$

Introducing (2.3.118) and (2.3.119) into (2.3.109), substituting the solution for Z(t) from (2.3.10), and simplifying, one can write the solution for x(t) as

$$x(t) = Z_{ss}[a_1(t) + a_2(t) - a_3(t)] + b_1(t) - b_2(t) + b_3(t) \ , \tag{2.3.120}$$

where

$$a_1(t) = e^{-\bar{A}^T(t-t_o)} [Z^{-1}(t_o)x_o - \xi(t_o)] \ ,$$

$$a_2(t) = G(-\bar{A}^T, K_2, \Omega, t, t_o)s_o \ ,$$

$$a_3(t) = G(-\bar{A}^T, K_4, L, t, t_o)\beta_o \ ,$$

$$b_1(t) = e^{\bar{A}(t-t_o)} [x_o - Z_{ss}(Z^{-1}(t_o)x_o - \xi(t_o))] \ ,$$

$$b_2(t) = G(\bar{A}, K_1, \Omega, t, t_o)s_o \ ,$$

$$b_3(t) = G(\bar{A},K_3,L,t,t_o)\beta_o ,$$

$G(A,B,C,t_2,t_1)$ is defined following (2.3.26), K_1 and K_2 are defined following (2.3.102), and K_3 and K_4 are defined following (2.3.104). The propagating equations for $a_i(t)$ and $b_i(t)$, i=1,2,3, can be shown to be:

$$a_1(t+\Delta t) = e^{-\bar{A}^T\Delta t}a_1(t) , \tag{2.3.121}$$

$$a_2(t+\Delta t) = e^{-\bar{A}^T\Delta t}a_2(t) + C_2 s(t) , \tag{2.3.122}$$

$$a_3(t+\Delta t) = e^{-\bar{A}^T\Delta t}a_3(t) + C_4\beta(t) , \tag{2.3.123}$$

$$b_1(t+\Delta t) = e^{\bar{A}\Delta t}b_1(t) , \tag{2.3.124}$$

$$b_2(t+\Delta t) = e^{\bar{A}\Delta t}b_2(t) + C_1 s(t) , \tag{2.3.125}$$

$$b_3(t+\Delta t) = e^{\bar{A}\Delta t}\, b_3(t) + C_3\beta(t) , \tag{2.3.126}$$

where

$$C_1 = G(\bar{A},K_1,\Omega,\Delta t,0) ,$$

$$C_2 = G(-\bar{A}^T,K_2,\Omega,\Delta t,0) ,$$

$$C_3 = G(\bar{A},K_3,L,\Delta t,0) ,$$

$$C_4 = G(-\bar{A}^T,K_4,L,\Delta t,0) ,$$

and the algorithm for computing the C_i coefficients is presented in Appendix B.

Treating the propagation equations for $a_i(t)$ and $b_i(t)$, i=1,2,3, as linear difference equations for discrete time systems, one notices that the system dynamics matrix for each of the discrete $a_i(t_k)$, namely $e^{-\bar{A}^T\Delta t}$, are the same; the system dynamics matrix for each of the discrete $b_i(t_k)$, namely $e^{\bar{A}\Delta t}$, are also the same. One can thus take advantage of linear superposition and re-write the solution for x(t) as

$$x(t) = Z_{ss}a(t) + b(t) \tag{2.3.127}$$

where

$$a(t) = a_1(t) + a_2(t) - a_3(t) ,$$

and

$$b(t) = b_1(t) - b_2(t) + b_3(t) .$$

The propagation equations for the new functions a(t) and b(t) can then be shown to be

$$a(t+\Delta t) = e^{-\bar{A}^T\Delta t}a(t) + C_2s(t) - C_4\beta(t) , \tag{2.3.128}$$

and

$$b(t+\Delta t) = e^{\bar{A}\Delta t}b(t) - C_1s(t) + C_3\beta(t) , \tag{2.3.129}$$

where the initial conditions for a(t) and b(t) are

$$a(t_o) = Z^{-1}(t_o)x_o - \xi(t_o) ,$$

and

$$b(t_o) = x_o - Z_{ss}[Z^{-1}(t_o)x_o - \xi(t_o)] ,$$

and the reference state and disturbance state are propagated via

$$s(t+\Delta t) = e^{\Omega \Delta t} s(t) , \tag{2.3.130}$$

and

$$\beta(t+\Delta t) = e^{L\Delta t}\beta(t) . \tag{2.3.131}$$

Note that the form of the solution for x(t) in (2.3.127) is identical to those derived for the linear optimal regulator ((2.3.18)) and the optimal controller with terminal constraints ((2.3.72)). The propagation equations for a(t) and b(t) are also similar, except that there are additional forcing terms for the tracking/disturbance accommodating controller. The propagation of a(t), b(t), s(t), and $\beta(t)$, and the multiplication of Z_{ss} and a(t) in (2.3.127) require a total of roughly $3n^2 + 4n_o^2 + n_d^2 + 4nn_o + 2nn_d$ flops per time-step, where the dimension of $\beta(t)$ is n_d and the dimension of s(t) is $2n_o$.

2.3.3.4 Control Trajectory Solution

The solution for the control trajectory may be obtained from (2.3.97) by direct substitution of the solutions for x(t) and $\xi(t)$. Let us first re-write the solution for $\xi(t)$ in terms of $a_1(t)$ and $b_1(t)$:

$$\xi(t) = Z^{-1}(t)e^{\bar{A}(t-t_o)}Z(t_o)\xi(t_o) + Z^{-1}(t)[b_3(t) - b_2(t)]$$

$$+ Z^{-1}(t)[Z(t) - Z_{ss}][a_3(t) - a_2(t)] . \tag{2.3.132}$$

By adding and subtracting terms of the form $Z^{-1}(t)b_1(t)$ and $Z^{-1}[Z(t) - Z_{ss}]a_1(t)$ to the above equation, $\xi(t)$ can be simplified to the form

$$\xi(t) = Z^{-1}(t)[b(t) - (Z(t) - Z_{ss})a(t)] . \qquad (2.3.133)$$

Substituting (2.3.10), (2.3.127) and (2.3.133) into (2.3.97) leads to

$$u(t) = -R^{-1}B^T[(P_{ss} + Z^{-1}(t))(Z_{ss}a(t) + b(t))$$

$$- Z^{-1}(t)b(t) + Z^{-1}(t)(Z(t) - Z_{ss})a(t)] , \qquad (2.3.134)$$

which can be simplified to

$$u(t) = -D_1 a(t) - D_2 b(t) , \qquad (2.3.135)$$

where

$$D_1 = R^{-1}B^T[P_{ss}Z_{ss} + I] ,$$

and

$$D_2 = R^{-1}B^T P_{ss} .$$

The equation for u(t) shown in (2.3.135) has the same form as those derived for the optimal linear regulator and optimal controller with terminal constraints. Evaluation of u(t) requires approximately 2pn flops per time-step.

2.3.3.5 Residual State Trajectory Solution

The residual state dynamics is governed by

$$\dot{x}_r(t) = A_r x_r(t) + B_r u(t) + \Lambda_r \beta(t) , \qquad (2.3.136)$$

where Λ_r is the $(n_r \times n_d)$ residual disturbance influence matrix, and x_r, A_r, and B_r are as defined in Section 2.3.1.1. On substitution of (2.3.135) into (2.3.136), the formal solution for $x_r(t)$ becomes

$$x_r(t) = e^{A_r(t-t_o)} x_r(t_o) - \int_{t_o}^{t} e^{A_r(t-\tau)} B_r D_1 a(\tau) d\tau$$

$$- \int_{t_o}^{t} e^{A_r(t-\tau)} B_r D_2 b(\tau) d\tau + \int_{t_o}^{t} e^{A_r(t-\tau)} \Lambda_r \beta(\tau) d\tau \ . \qquad (2.3.137)$$

Using the expressions for a(t), b(t) and $\beta(t)$ from (2.3.127), (2.3.120), and (2.3.86), one can re-write (2.3.137) as

$$x_r(t) = e^{A_r(t-t_o)} x_r(t_o) - G(A_r, B_r D_1, -\bar{A}^T, t, t_o) a(t_o)$$

$$- G(A_r, B_r D_2, \bar{A}, t, t_o) b(t_o)$$

$$- [\, H(A_r, B_r D_1, -\bar{A}^T, K_2, \Omega, t, t_o) - H(A_r, B_r D_2, \bar{A}, K_1, \Omega, t, t_o) \,] s_o$$

$$+ [\, G(A_r, \Lambda_r, L, t, t_o) + H(A_r, B_r D_1, -\bar{A}^T, K_4, L, t, t_o)$$

$$- H(A_r, B_r D_2, \bar{A}, K_3, L, t, t_o) \,] \beta_o \ , \qquad (2.3.138)$$

where

$$H(A, B, C, D, E, t_2, t_1) = \int_{t_1}^{t_2} e^{A(t_2-\tau)} B \int_{t_1}^{\tau} e^{C(\tau-\varepsilon)} D e^{E(\varepsilon-t_1)} d\varepsilon \, d\tau \ ,$$

B and D are general constant matrices, vectors, or scalars, and A, C, and E are general square matrices or scalars.

As shown in Appendix E, the solution for $x_r(t)$ may be propagated at intervals of Δt by the following difference equation:

$$x_r(t+\Delta t) = e^{A_r \Delta t} x_r(t) - D_3 a(t) - D_4 b(t) + D_5 s(t) + D_6 \beta(t) , \tag{2.3.139}$$

where

$$D_3 = G(A_r, B_r D_1, -\bar{A}^T, \Delta t, 0) ,$$

$$D_4 = G(A_r, B_r D_2, \bar{A}, \Delta t, 0) ,$$

$$D_5 = H(A_r, B_r D_2, \bar{A}, K_1, \Omega, \Delta t, 0) - H(A_r, B_r D_1, -\bar{A}^T, K_2, \Omega, \Delta t, 0) ,$$

and

$$D_6 = G(A_r, \Lambda_r, L, \Delta t, 0) + H(A_r, B_r D_1, -\bar{A}^T, K_4, L, \Delta t, 0)$$

$$- H(A_r, B_r D_2, \bar{A}, K_3, L, \Delta t, 0) .$$

Assuming $a(t)$, $b(t)$, $s(t)$, and $\beta(t)$ are already computed for the state trajectory propagation, the above computation for $x_r(t)$ requires about $n_r^2 + 2nn_r + 2n_o n_r + n_r n_d$ flops per time-step.

If a control smoothing formulation is used, where the state vector is augmented by the control and control-rate vectors, the residual dynamics equation is modified to be

$$\dot{x}_r(t) = A_r x_r(t) + B_r H_u x(t) + \Lambda_r \beta(t) , \tag{2.3.140}$$

where H_u is a selection matrix for the control vector portion of the augmented state. The solution for $x_r(t)$ is then written as

$$x_r(t) = e^{A_r(t-t_o)} x_r(t_o)$$

$$+ G(A_r, B_r H_u Z_{ss}, -\bar{A}^T, t, t_o) a(t_o)$$

$$+ G(A_r, B_r H_u, \bar{A}, t, t_o) b(t_o)$$

$$+ [\, H(A_r, B_r H_u Z_{ss}, -\bar{A}^T, K_2, \Omega, t, t_o) - H(A_r, B_r H_u, \bar{A}, K_1, \Omega, t, t_o) \,] s_o$$

$$+ [\, G(A_r, \Lambda_r, L, t, t_o) - H(A_r, B_r H_u Z_{ss}, -\bar{A}^T, K_4, L, t, t_o)$$

$$+ H(A_r, B_r H_u, \bar{A}, K_3, L, t, t_o) \,] \beta_o \,, \qquad (2.3.141)$$

and the propagation equation then becomes

$$x_r(t+\Delta t) = e^{A_r \Delta t} x_r(t) + D_3' a(t) + D_4' b(t) + D_5' s(t) + D_6' \beta(t) \,, \qquad (2.3.142)$$

where

$$D_3' = G(A_r, B_r H_u Z_{ss}, -\bar{A}^T, \Delta t, 0) \,,$$

$$D_4' = G(A_r, B_r H_u, \bar{A}, \Delta t, 0) \,,$$

$$D_5' = H(A_r, B_r H_u Z_{ss}, -\bar{A}^T, K_2, \Omega, \Delta t, 0) - H(A_r, B_r H_u, \bar{A}, K_1, \Omega, \Delta t, 0) \,,$$

and

$$D_6' = G(A_r, A_r, L, \Delta t, 0) - H(A_r, B_r H_u Z_{ss}, -\bar{A}^T, K_4, L, \Delta t, 0)$$

$$+ H(A_r, B_r H_u, \bar{A}, K_3, L, \Delta t, 0) .$$

The total amount of work required to compute the state, control, and residual state trajectories for the optimal tracking/disturbance accommodating control problem is approximately $3n^2 + n_r^2 + 4n_o^2 + n_d^2 + 4nn_o + 2nn_d + 2nn_r + 2n_on_r + n_rn_d + 2pn$ flops per time-step.

2.3.3.6 Comparison with Alternative Closed-Form Solution

The following open-loop solution to the optimal tracking/disturbance accommodating control problem is presented for comparison. In order to compute the unknown initial costates, the state, costate, reference state, and disturbance state may be put together into a higher-dimensioned system state governed by the differential equation

$$\begin{Bmatrix} \dot{x}(t) \\ \dot{\lambda}(t) \\ \dot{s}(t) \\ \dot{\beta}(t) \end{Bmatrix} = \begin{bmatrix} A & -E & 0 & \Lambda \\ -Q & -A^T & \tilde{Q}H_x & 0 \\ 0 & 0 & \Omega & 0 \\ 0 & 0 & 0 & L \end{bmatrix} \begin{Bmatrix} x(t) \\ \lambda(t) \\ s(t) \\ \beta(t) \end{Bmatrix} . \qquad (2.3.143)$$

Denoting the matrix coefficient of the above matrix by K, one can write the final time transition matrix as

$$e^{K(t_f - t_o)} = \begin{bmatrix} \Theta_{11} & \Theta_{12} & \Gamma_{11} & \Gamma_{12} \\ \Theta_{21} & \Theta_{22} & \Gamma_{21} & \Gamma_{22} \\ 0 & 0 & e^{\Omega(t_f - t_o)} & 0 \\ 0 & 0 & 0 & e^{L(t_f - t_o)} \end{bmatrix} . \qquad (2.3.144)$$

The transition matrix partitions can then be used to write expressions for

the variables appearing in the terminal condition for the costate (see (2.3.94)):

$$x(t_f) = \Theta_{11}x_o + \Theta_{12}\lambda(t_o) + \Gamma_{11}s_o + \Gamma_{12}\beta_o , \tag{2.3.145}$$

$$\lambda(t_f) = \Theta_{21}x_o + \Theta_{22}\lambda(t_o) + \Gamma_{21}s_o + \Gamma_{22}\beta_o , \tag{2.3.146}$$

and

$$s(t_f) = e^{\Omega(t_f-t_o)}s_o . \tag{2.3.147}$$

Substituting (2.3.145) through (2.3.147) into (2.3.94) and solving for $\lambda(t_o)$ leads to

$$[\Theta_{22} - P_f\Theta_{12}]\lambda(t_o) = [P_f\Theta_{11} - \Theta_{21}]x_o + [P_f\Gamma_{11} - \Gamma_{21}]s_o + [P_f\Gamma_{12} - \Gamma_{22}]\beta_o - \tilde{P}_f H_x e^{\Omega(t_f-t_o)}s_o , \tag{2.3.148}$$

from which $\lambda(t_o)$ can be obtained via Gaussian elimination.

In order to obtain the state, control, and residual mode trajectories, the system state of (2.3.143) is augmented to the form

$$\begin{Bmatrix} \dot{x}_r(t) \\ \dot{x}(t) \\ \dot{\lambda}(t) \\ \dot{s}(t) \\ \dot{\beta}(t) \end{Bmatrix} = \begin{bmatrix} A_r & 0 & -E_r & 0 & \Lambda_r \\ 0 & A & -E & 0 & \Lambda \\ 0 & -Q & -A^T & \tilde{Q}H_x & 0 \\ 0 & 0 & 0 & \Omega & 0 \\ 0 & 0 & 0 & 0 & L \end{bmatrix} \begin{Bmatrix} x_r(t) \\ x(t) \\ \lambda(t) \\ s(t) \\ \beta(t) \end{Bmatrix} . \tag{2.3.149}$$

Denoting the matrix coefficient of the above equation as K, one can write the single step transition matrix in the form

$$e^{K\Delta t} = \begin{bmatrix} e^{A_r\Delta t} & \Theta_{r1} & \Theta_{r2} & \Gamma_{r1} & \Gamma_{r2} \\ 0 & \Theta_{11} & \Theta_{12} & \Gamma_{11} & \Gamma_{12} \\ 0 & \Theta_{21} & \Theta_{22} & \Gamma_{21} & \Gamma_{22} \\ 0 & 0 & 0 & e^{\Omega\Delta t} & 0 \\ 0 & 0 & 0 & 0 & e^{L\Delta t} \end{bmatrix} . \qquad (2.3.150)$$

The propagation equation for the total system becomes

$$\begin{Bmatrix} x_r(t+\Delta t) \\ x(t+\Delta t) \\ \lambda(t+\Delta t) \\ s(t+\Delta t) \\ \beta(t+\Delta t) \end{Bmatrix} = e^{K\Delta t} \begin{Bmatrix} x_r(t) \\ x(t) \\ \lambda(t) \\ s(t) \\ \beta(t) \end{Bmatrix} , \qquad (2.3.151)$$

where the initial costates are given by (2.3.148), and the equation for the control is given by (2.3.95). The propagation of the total system state and the calculation of the control require $4n^2 + n_r^2 + 4n_o^2 + n_d^2 + 4nn_o + 2nn_d + 2nn_r + 2n_on_r + n_rn_d + pn$ flops per time-step. This estimate is more than that shown for the Z(t) type solutions if $p < n$. For the case where there are as many controls as states, (p=n), the two work estimates are the same.

For the case where the plant states x(t) are augmented to include control states, the residual state dynamics shown in (2.3.149) are modified as follows:

$$\dot{x}_r(t) = A_r x_r(t) + B_r H_u x(t) + \Lambda_r \beta(t) , \qquad (2.3.152)$$

where H_u is a selection matrix for the control partition of the augmented state. The remaining calculations are the same as indicated in (2.3.150) and (2.3.151).

Note that for the tracking/disturbance accommodating controller, if one sets $\beta(t) = 0$, the equations of this subsection reduce to those for a tracking controller, and if one sets $s(t) = 0$, the equations correspond to those for a disturbance accommodating controller. If $\beta(t)$ and $s(t)$ are both set to zero, the equations become identical to those presented in Section 2.3.1 for the optimal linear regulator. Setting $\beta(t)$ to a constant vector leads to a tracking controller subject to a constant disturbance. Similarly, setting $s(t)$ to a constant vector allows one to track a constant reference state subject to deterministic disturbances. Numerical results for the tracking/disturbance accommodating controller are shown in Ref. [20].

2.3.3.7 Propagation of the Feedback Gains

For the actual implementation of the tracking/disturbance accommodating controller, the Riccati matrix, $P(t)$, and the feedforward vector, $\xi(t)$, need to be propagated. Since the solution for $P(t)$ is the same as that for the linear optimal regulator, the propagation for $P(t)$ is performed by the same propagation and inversion of $Z(t)$ as discussed in Section 2.3.1.2. The propagation equation for $\xi(t)$ is obtained by re-writing (2.3.133) in the form

$$\xi(t+\Delta t) = Z^{-1}(t+\Delta t)\left[b(t+\Delta t) - \left(Z(t+\Delta t) - Z_{ss}\right)a(t+\Delta t)\right] , \quad (2.3.153)$$

where $Z(t+\Delta t)$, $a(t+\Delta t)$, and $b(t+\Delta t)$ are computed via (2.3.11), (2.3.128), and (2.3.129), respectively. The disturbance and reference states, which are needed in the propagation of $a(t)$ and $b(t)$, are obtained via (2.3.130) and (2.3.131). By performing the subtractions in the brackets before the products, and assuming that $Z(t+\Delta t)$ and its Cholesky decomposition are already available from the propagation of $P(t)$, the operations indicated by (2.3.153) require roughly $2n^2$ flops. The propagation for $a(t)$ and $b(t)$ shown in (2.3.128) and (2.3.129) require $n^2 + 2nn_o + nn_d$ flops each, while the propagation of $s(t)$ and $\beta(t)$ shown in (2.3.130) and (2.3.131) require $4n_o^2$ and n_d^2 flops, respectively. Thus, assuming that $Z(t+\Delta t)$ has been computed and decomposed, the amount of work required for propagating $\xi(t)$,

including a(t), b(t), s(t) and $\beta(t)$, is about $4n^2 + 4n_o^2 + n_d^2 + 4nn_o + 2nn_d$ flops.

For comparison, a Kalman-Englar type of closed-form solution is presented for the propagation of $\xi(t)$. Replacing the time argument of (2.3.144) by $t+\Delta t$, one can write the state and costate propagating equations as

$$x(t+\Delta t) = \theta_{11}x(t) + \theta_{12}\lambda(t) + \Gamma_{11}s(t) + \Gamma_{12}\beta(t) , \qquad (2.3.154)$$

and

$$\lambda(t+\Delta t) = \theta_{21}x(t) + \theta_{22}\lambda(t) + \Gamma_{21}s(t) + \Gamma_{22}\beta(t) , \qquad (2.3.155)$$

where $\theta_{ij} = \theta_{ij}(t+\Delta t,t)$ and $\Gamma_{ij} = \Gamma_{ij}(t+\Delta t,t)$ are partitions of the single-step transition matrix. Replacing $\lambda(t)$ of (2.3.154) and (2.3.155) by $P(t)x(t) - \xi(t)$ from (2.3.96), and eliminating x(t), one obtains

$$\lambda(t+\Delta t) = P(t+\Delta t)x(t+\Delta t) - [\theta_{22} - P(t+\Delta t)\theta_{12}]\xi(t)$$

$$- [P(t+\Delta t)\Gamma_{11} - \Gamma_{21}]s(t) - [P(t+\Delta t)\Gamma_{12} - \Gamma_{22}]\beta(t) , \qquad (2.3.156)$$

where

$$P(t+\Delta t) = [\theta_{21} + \theta_{22}P(t)][\theta_{11} + \theta_{12}P(t)]^{-1} .$$

Comparison of (2.3.156) with (2.3.96) leads to the Kalman-Englar type of propagation for $\xi(t)$:

$$\xi(t+\Delta t) = [\theta_{22} - P(t+\Delta t)\theta_{12}]\xi(t) + [P(t+\Delta t)\Gamma_{11} - \Gamma_{21}]s(t)$$

$$+ [P(t+\Delta t)\Gamma_{12} - \Gamma_{22}]\beta(t) . \qquad (2.3.157)$$

Re-arrangement of the above equation leads to a more efficient propagation:

$$\xi(t+\Delta t) = -P(t+\Delta t)[\theta_{12}\xi(t) - \Gamma_{11}s(t) - \Gamma_{12}\beta(t)]$$

$$+ \theta_{22}\xi(t) - \Gamma_{21}s(t) - \Gamma_{22}\beta(t) . \qquad (2.3.158)$$

By performing the operations within the brackets first, and assuming that $P(t+\Delta t)$ is already computed, the propagation of $\xi(t)$ via (2.3.158) requires roughly $3n^2 + 4n_o^2 + n_d^2 + 4nn_o + 2nn_d$ flops, including the amount of computation needed for propagating $s(t)$ and $\beta(t)$ via (2.3.130) and (2.3.131). This operation count for propagating $\xi(t)$ is lower than for the $Z(t)$ type solution. However, this difference is not significant when compared to the $O(n^3)$ difference in the propagation of $P(t)$, which favors the $Z(t)$ type solution. (See Section 2.3.1.2). For the least amount of work, one may propagate $P(t)$ using Potter's solution, and propagate $\xi(t)$ using the Kalman-Englar type solution of (2.3.158). However, the numerical stability of using (2.3.158) has not yet been tested nor studied.

2.3.4 Subspace Reduction for the Hamiltonian Matrix

In Sections 2.3.1 through 2.3.3, comparisons were made between $Z(t)$ type solutions involving variables such as P_{ss}, Z_{ss}, $e^{\bar{A}t}$, and $e^{-\bar{A}^T t}$, and open-loop solutions involving partitions of the exponentials of the Hamiltonian matrix. This section shows the connection between the two types of solutions by means of subspace reductions.

Given the Hamiltonian matrix

$$\Omega = \begin{bmatrix} A & -E \\ -Q & -A^T \end{bmatrix} , \qquad (2.3.159)$$

where A, E, and Q are as defined in Section 2.3.1.1, let us apply

similarity transformations which transform the Hamiltonian matrix into a block diagonal form. In particular, let us consider the following reducing subspace transformations:

$$\begin{bmatrix} I & 0 \\ P_1 & I \end{bmatrix}^{-1} = \begin{bmatrix} I & 0 \\ -P_1 & I \end{bmatrix}, \tag{2.3.160}$$

and

$$\begin{bmatrix} I & P_2 \\ 0 & I \end{bmatrix}^{-1} = \begin{bmatrix} I & -P_2 \\ 0 & I \end{bmatrix}. \tag{2.3.161}$$

Note that the above transformations are symplectic if $P_1 = P_1^T$ and $P_2 = P_2^T$. Applying the first transformation to the Hamiltonian matrix produces

$$\begin{bmatrix} I & 0 \\ -P_1 & I \end{bmatrix} \begin{bmatrix} A & -E \\ -Q & -A^T \end{bmatrix} \begin{bmatrix} I & 0 \\ P_1 & I \end{bmatrix} = \begin{bmatrix} A-EP_1 & -E \\ -P_1A-A^TP_1+P_1EP_1-Q & P_1E-A^T \end{bmatrix}, \tag{2.3.162}$$

It is easy to observe that the condition for setting the lower left partition of (2.3.162) to zero is equivalent to solving the steady-state Riccati equation. Thus, on setting $P_1 = P_{ss}$, the matrix in (2.3.162) becomes

$$\begin{bmatrix} \bar{A} & -E \\ 0 & -\bar{A}^T \end{bmatrix},$$

where $\bar{A} = A - EP_{ss}$. Similarly, application of the second transformation leads to

$$\begin{bmatrix} I & -P_2 \\ 0 & I \end{bmatrix} \begin{bmatrix} \bar{A} & -E \\ 0 & -\bar{A}^T \end{bmatrix} \begin{bmatrix} I & P_2 \\ 0 & I \end{bmatrix} = \begin{bmatrix} \bar{A} & \bar{A}P_2+P_2\bar{A}^T-E \\ 0 & -\bar{A}^T \end{bmatrix}, \tag{2.3.163}$$

The condition for the upper right partition of (2.3.163) to be zero is the

steady-state Lyapunov equation for Z(t) following (2.2.10). On setting $P_2 = Z_{ss}$, the resulting block diagonal matrix becomes

$$\begin{bmatrix} \tilde{A} & 0 \\ 0 & -\tilde{A}^T \end{bmatrix}.$$

Note that exponentials of the above matrix multiplied by a time function are symplectic, which is also a property of the state transition matrix. By multiplying the two transformations together, one can show that the Hamiltonian matrix is related to the $\tilde{A}$ matrix by the following transformation:

$$\begin{bmatrix} A & -E \\ -Q & -A^T \end{bmatrix} = \begin{bmatrix} I & Z_{ss} \\ P_{ss} & P_{ss}Z_{ss}+I \end{bmatrix} \begin{bmatrix} \tilde{A} & 0 \\ 0 & -\tilde{A}^T \end{bmatrix} \begin{bmatrix} I+Z_{ss}P_{ss} & -Z_{ss} \\ -P_{ss} & I \end{bmatrix}. \tag{2.3.164}$$

It can be shown that the transformation matrix of the above equation is symplectic. Also, the fact that the Hamiltonian has pairs of eigenvalues of opposite sign easily follows.

The advantage of a block diagonal form for the Hamiltonian is that in forming the state transition matrix, one exponentiates two matrices of order n rather than one matrix of order 2n. Since the work required for matrix exponentiation is roughly proportional to the third power of the matrix dimension, this represents a substantial computational benefit.

On exponentiating (2.3.164) with a product of time, one observes that the partitions of the state transition matrix can be expressed entirely in terms of P_{ss}, Z_{ss}, $e^{\tilde{A}t}$, and $e^{-\tilde{A}^T t}$. Indeed, these are the same terms which appear in the Z(t) type solutions. Thus the Z(t) type solutions for the optimal controllers may be viewed as reduced subspace forms of the transition matrix solutions.

2.3.5 Summary

It is interesting to note that the optimal state and control trajectories for the optimal linear regulator, optimal controller with terminal constraints, and optimal tracking/disturbance accommodating controller all have the same form, namely:

$$x(t) = Z_{ss}a(t) + b(t) , \tag{2.3.165}$$

and

$$u(t) = - D_1 a(t) - D_2 b(t) , \tag{2.3.166}$$

where

$$D_1 = R^{-1}B^T[P_{ss}Z_{ss} + I] ,$$

and

$$D_2 = R^{-1}B^T P_{ss} .$$

Moreover, the propagation equations for the state, control and residual state for all three controllers may be viewed as solutions to the difference equations for equivalent discrete-time control systems.

Another useful observation is that in addition to obtaining residual state trajectories, the residual state propagation equations may be used to simulate the response of an off-nominal plant when subjected to the nominal open-loop controls. This is done by simply replacing A_r, B_r, and Λ_r by A_p, B_p, and Λ_p, where the subscript p indicates the perturbed plant.

Table 2-3 provides a summary of the solutions to the three types of controllers.

Table 2-3. Summary of Example Applications of Closed-Form Solutions

Type of Controller	Optimal Linear Regulator	Controller with Terminal Constraints	Tracking/Disturbance Accommodating Controller
Performance Index	$J = \frac{1}{2}\|\|y(t_f)\|\|^2_P + \frac{1}{2}\int_{t_o}^{t_f}[\|\|y(t)\|\|^2_Q + \|\|u(t)\|\|^2_R]dt$	$J = \frac{1}{2}\|\|y(t_f)\|\|^2_P + \frac{1}{2}\int_{t_o}^{t_f}[\|\|y(t)\|\|^2_Q + \|\|u(t)\|\|^2_R]dt$	$J = \frac{1}{2}\|\|y^*(t_f) - y(t_f)\|\|^2_P + \frac{1}{2}\int_{t_o}^{t_f}[\|\|y^*(t) - y(t)\|\|^2_Q + \|\|u(t)\|\|^2_R]dt$
Necessary Conditions	$\dot{x}(t) = Ax(t) + Bu(t)$, x_o given $\dot{\lambda}(t) = -Qx(t) - A^T\lambda(t)$, $\lambda(t_f) = P_f x(t_f)$ $u(t) = -R^{-1}B^T\lambda(t)$	$\dot{x}(t) = Ax(t) + Bu(t)$, x_o given $\dot{\lambda}(t) = -Qx(t) - A^T\lambda(t)$, $\lambda(t_f) = C^T M^T \upsilon + P_f x(t_f)$ $u(t) = -R^{-1}B^T\lambda(t)$ $\psi(t_f) = MCx(t_f) - \psi_d = 0$	$\dot{x}(t) = Ax(t) + Bu(t) + \Lambda\beta(t)$, x_o given $\dot{\lambda}(t) = -Qx(t) + \tilde{Q}H_x s(t) - A^T\lambda(t)$, $\lambda(t_f) = P_f x(t_f) - \tilde{P}_f H_x s(t_f)$ $u(t) = -R^{-1}B^T\lambda(t)$
Feedback Form	$\lambda(t) = P(t)x(t)$	$\lambda(t) = P(t)x(t) - S(t)\psi_d$	$\lambda(t) = P(t)x(t) - \xi(t)$
Sweep Variables	$\dot{P} = -PA - A^TP + PEP - Q$, $P(t_f) = P_f$	$\dot{P} = -PA - A^TP + PEP - Q$, $P(t_f) = \bar{P} - \bar{S}\bar{G}^{-1}\bar{S}^T$ $\dot{S} = -[A - EP]^T S$, $S(t_f) = -\bar{S}\bar{G}^{-1}$	$\dot{P} = -PA - A^TP + PEP - Q$, $P(t_f) = P_f$ $\dot{\xi} = -[A - EP]^T\xi - \tilde{Q}H_x s + P\Lambda\beta$, $\xi(t_f) = \tilde{P}_f H_x s(t_f)$
State Trajectory	$x(t) = Z_{ss}a(t) + b(t)$	$x(t) = Z_{ss}a(t) + b(t)$	$x(t) = Z_{ss}a(t) + b(t)$
Control Trajectory	$u(t) = -D_1 a(t) - D_2 b(t)$	$u(t) = -D_1 a(t) - D_2 b(t)$	$u(t) = -D_1 a(t) - D_2 b(t)$

Table 2-3. Summary of Example Applications of Closed-Form Solutions (Continued)

Expression for a(t)	$a(t) = e^{-\bar{A}^T(t-t_o)} Z^{-1}(t_o)x_o$	$a(t) = e^{-\bar{A}^T(t-t_o)}[Z^{-1}(t_o)x_o - S(t_o)\psi_d]$	$a(t) = e^{-\bar{A}^T(t-t_o)}[Z^{-1}(t_o)x_o - \xi(t_o)]$ $+ G(-\bar{A}^T, K_2, \Omega, t, t_o)s_o$ $- G(-\bar{A}^T, K_4, L, t, t_o)\beta_o$
Expression for b(t)	$b(t) = e^{\bar{A}(t-t_o)}[Z(t_o) - Z_{ss}]$ $\times Z^{-1}(t_o)x_o$	$b(t) = e^{\bar{A}(t-t_o)}$ $\times [x_o - Z_{ss}(Z^{-1}(t_o)x_o - S(t_o)\psi_d)]$	$b(t) = e^{\bar{A}(t-t_o)}[x_o - Z_{ss}(Z^{-1}(t_o)x_o - \xi(t_o))]$ $- G(\bar{A}, K_1, \Omega, t, t_o)s_o$ $+ G(\bar{A}, K_3, L, t, t_o)\beta_o$
Recursion for a(t)	$a(t + \Delta t) = e^{-\bar{A}^T\Delta t}a(t)$	$a(t + \Delta t) = e^{-\bar{A}^T\Delta t}a(t)$	$a(t + \Delta t) = e^{-\bar{A}^T\Delta t}a(t) + C_2 s(t) - C_4\beta(t)$
Recursion for b(t)	$b(t + \Delta t) = e^{\bar{A}\Delta t}b(t)$	$b(t + \Delta t) = e^{\bar{A}\Delta t}b(t)$	$b(t + \Delta t) = e^{\bar{A}\Delta t}b(t) - C_1 s(t) + C_3\beta(t)$
Work for x(t) and u(t) propagation (flops)	$3n^2 + 2pn$	$3n^2 + 2pn$	$3n^2 + 2pn + 4n_o^2 + n_d^2 + 4nn_o + 2nn_d$
Work for transition matrix solution to x(t) and u(t) (flops)	$4n^2 + pn$	$4n^2 + pn$	$4n^2 + pn + 4n_o^2 + n_d^2 + 4nn_o + 2nn_d$

2.4 Conclusions

This chapter has dealt with a new class of closed-form solutions for finite-time linear-quadratic optimal control problems. Through illustrative examples, this class of optimal control solutions is found to require less computational work than the solutions based on the state transition matrix. Moreover, these solutions involve the exponentiation of matrices which are of lower order than those needed for the transition matrix method. Since the amount of work required to exponentiate a matrix is proportional to the third power of the matrix dimension, and since a higher matrix dimension may potentially adversely affect the accuracy of the resulting matrix exponential, the class of solutions involving Z(t) is very useful when the number of state variables is large.

Furthermore, numerical problems exist when the feedback gains are propagated using partitions of the state transition matrix with a Kalman-Englar type of approach. Vaughan's negative exponential solution solves the problem of numerical instability by using the eigenvectors of the transition matrix; however, since this involves complex arithmetic, the computational work required for propagating the gains is many times higher than that for Kalman-Englar type solutions. In contrast, the amount of work required for propagating the feedback gains using the Z(t) type solutions is less than that needed for Kalman-Englar type solutions, and moreover, the method seems to be numerically stable. However, analysis of the numerical stability and error propagation characteristics for the Z(t) type solutions remains a topic for further research.

CHAPTER 3

SPACECRAFT SLEWING MANEUVERS USING A CLOSED-FORM SOLUTION FOR THE NEIGHBORING EXTREMAL PATH PROBLEM

3.1 Introduction

The optimal control problem in this chapter is specified by defining a performance index which consists of a penalty on elapsed time, a quadratic penalty on the terminal states and controls, and an integral of quadratic penalties on the states, controls, and control rates. The final time is free, and specified terminal constraints produce a terminal manifold which also may be a function of the final time. Assuming that the nominal control and state trajectories are known, one seeks the perturbation feedback gains which cause the system to follow a neighboring extremal path when subjected to small perturbations in the initial conditions and terminal constraints. Necessary conditions for the perturbed system are stated, and the solution for the nominal trajectory is shown. Solutions for the perturbation feedback gains are developed based on the results of Chapter 2. Perfect plant knowledge and perfect state estimation is assumed. Computationally efficient propagation equations are shown for computing the feedback gains at fixed time intervals. A time-to-go indexing scheme is used for applying the feedback gains so that the controller does not run out of feedback gains if the actual final time is longer than the nominal final time. Slight numerical modifications are presented for overcoming the numerical sensitivities of this type of controller. Four spinup and retargeting example maneuvers are shown, involving a spacecraft model consisting of a rigid body with four flexible appendages. An extension is proposed for using the closed-form solutions in control problems involving nonlinear systems by linearizing the nonlinear plant equations about the nominal trajectory.

3.2 Statement of the Control Problem

Let us assume that we have obtained the p-dimensional nominal

control vector $u^N(t)$, which minimizes the quadratic performance index

$$J = W_t t_f + \frac{1}{2} x_f^T S_f x_f + \frac{1}{2} \int_{t_o}^{t_f} [x^T W_{xx} x + u^T W_{uu} u] dt , \qquad (3.2.1)$$

subject to

$$\dot{x} = Ax + Bu , \quad x(t_o) = x_o \text{ given} , \qquad (3.2.2)$$

$$\psi[x(t_f), t_f] = Mx(t_f) - \psi_d^N(t_f) = 0 , \quad \text{and } t_f \text{ unspecified} . \qquad (3.2.3)$$

In the above equations, x is the n-dimensional state vector, A and B are the time-invariant state dynamics and control influence matrices, ψ is a q-dimensional vector of terminal constraints, $S_f = S_f^T \geq 0$ and $W_{xx} = W_{xx}^T \geq 0$ are weighting matrices for the state, $W_{uu} = W_{uu}^T > 0$ is a weighting matrix for the control, and $W_t \geq 0$ is a weight for the final time. The following necessary conditions must be satisfied by the nominal optimal control and state trajectories [14]:

$$\dot{x} = Ax + Bu , \quad x(t_o) \text{ given} , \qquad (3.2.4)$$

$$\dot{\lambda} = - W_{xx} x - A^T \lambda , \quad \lambda(t_f) = S_f x(t_f)^T + M \nu , \qquad (3.2.5)$$

$$u = - W_{uu}^{-1} B^T \lambda , \qquad (3.2.6)$$

$$\psi[x(t_f), t_f] = 0 , \qquad (3.2.7)$$

$$\Omega = \frac{d\Phi}{dt}\Big|_{t_f} + \frac{1}{2} [x^T W_{xx} x + u^T W_{uu} u]\Big|_{t_f} = 0 , \qquad (3.2.8)$$

where

$$\Phi = W_t t_f + \frac{1}{2} x_f^T S_f x_f + \nu^T \psi ,$$

$$\frac{d\Phi}{dt} = W_t - \nu^T \psi_d^N(t_f) + [x_f^T S_f + \nu^T M]\dot{x}_f ,$$

$\lambda(t)$ is the n-dimensional costate vector, and ν is the q-dimensional vector of Lagrange multipliers for the terminal constraints.

Let us now consider small perturbations in the initial states $\delta x(t_o)$, and in the terminal constraints $d\psi$. The perturbation problem is then to seek the the correction to the control, $\delta u(t)$, which causes the perturbed system to minimize the original performance index subject to the new initial conditions and new final constraints. Moreover, we seek a feedback form for the solution of $\delta u(t)$, which involves the perturbations in the state, $\delta x(t)$, and the perturbations in the final conditions $d\psi$. The necessary conditions which must be satisfied by the perturbed system are given by the following equations [14]:

$$\delta\dot{x} = A\delta x + B\delta u, \quad \delta x(t_o) \text{ given} , \tag{3.2.9}$$

$$\delta\dot{\lambda} = - W_{xx}\delta x - A^T\delta\lambda , \tag{3.2.10}$$

$$\delta u = - W_{uu}^{-1} B^T \delta\lambda , \tag{3.2.11}$$

$$\delta\lambda(t_f) = \left[\frac{\partial^2\Phi}{\partial x^2}\right]\Big|_{t_f} \delta x_f + \left[\frac{\partial\psi}{\partial x}\right]\Big|_{t_f}^T d\nu + \left[\frac{\partial\Omega}{\partial x}\right]\Big|_{t_f}^T dt_f$$

$$= S_f\delta x + M^T d\nu + [S_f\dot{x}_f + A^T S_f x_f + A^T M^T \nu + W_{xx}x_f]dt_f , \tag{3.2.12}$$

$$d\psi = \left[\frac{\partial\psi}{\partial x}\right]\Big|_{t_f} \delta x_f + \left[\frac{d\psi}{dt}\right]\Big|_{t_f}^T dt_f = M\delta x_f + [M\dot{x}_f - \dot{\psi}_d^N(t_f)]dt_f ,$$

(3.2.13)

and

$$d\Omega = \left[\frac{\partial\Omega}{\partial x}\right]\Big|_{t_f} \delta x_f + \left[\frac{d\psi}{dt}\right]\Big|_{t_f}^{T} d\nu + \left[\frac{d\Omega}{dt}\right]\Big|_{t_f} dt_f = 0$$

$$= \left[\dot{x}_f^T S_f + x_f^T S_f A + \nu^T MA + x_f^T W_{xx}\right]\delta x_f + \left[M\dot{x}_f - \psi_d^N(t_f)\right]^T d\nu$$

$$+ \left[-\nu^T \ddot{\psi}_d(t_f) + (\dot{x}_f^T S_f + x_f^T S_f A + \nu^T MA + x_f^T W_{xx})\dot{x}_f\right]dt_f \ . \tag{3.2.14}$$

3.3 Solution for the Nominal Trajectory

The solution for the nominal trajectory may be obtained by using the state transition matrix and an exponential form for J. For a given final time t_f, the final states and costates can be written as

$$\begin{Bmatrix} x(t_f) \\ \lambda(t_f) \end{Bmatrix} = \left[e^{G(t_f - t_o)}\right] \begin{Bmatrix} x(t_o) \\ \lambda(t_o) \end{Bmatrix}, \tag{3.3.1}$$

where

$$G = \begin{bmatrix} A & -BW_{uu}^{-1}B^T \\ -W_{xx} & -A^T \end{bmatrix},$$

and

$$e^{G(t_f - t_o)} = \begin{bmatrix} \Phi_{xx} & \Phi_{x\lambda} \\ \Phi_{\lambda x} & \Phi_{\lambda\lambda} \end{bmatrix}$$

is the state transition matrix for $t = t_f$. By introducing the terminal constraint of (3.2.3) into (3.3.1), and using the boundary condition of (3.2.5), one obtains

$$\begin{bmatrix} I & 0 \\ S_f & M^T \\ M & 0 \end{bmatrix} \begin{Bmatrix} x(t_f) \\ \nu \end{Bmatrix} = \begin{bmatrix} \Phi_{xx} & \Phi_{x\lambda} \\ \Phi_{\lambda x} & \Phi_{\lambda\lambda} \\ 0 & 0 \end{bmatrix} \begin{Bmatrix} x(t_o) \\ \lambda(t_o) \end{Bmatrix} + \begin{Bmatrix} 0 \\ 0 \\ \psi_d^N(t_f) \end{Bmatrix} . \tag{3.3.2}$$

Upon rearranging and placing the unknown variables on the left hand side, we get

$$\begin{bmatrix} I & 0 & -\Phi_{x\lambda} \\ S_f & M^T & -\Phi_{\lambda\lambda} \\ M & 0 & 0 \end{bmatrix} \begin{Bmatrix} x(t_f) \\ \nu \\ \lambda(t_o) \end{Bmatrix} = \begin{bmatrix} \Phi_{xx} & 0 \\ \Phi_{\lambda x} & 0 \\ 0 & I \end{bmatrix} \begin{Bmatrix} x(t_o) \\ \psi_d^N(t_f) \end{Bmatrix} , \tag{3.3.3}$$

from which $x(t_f)$, ν, and $\lambda(t_o)$ may be obtained via Gaussian elimination.

The above solution assumes that t_f is known. However, for a free-final-time problem, the optimality condition, $\Omega = 0$, of (3.2.8) must be satisfied as well. This conditon produces a local minimum for J. To obtain the global minimum, one can numerically compute the value of J over a reasonable range of t_f and find which value of t_f produces the lowest value of J. Efficient propagation algorithms for J are of central importance in finding the minimum value of J. To find an expression for J we first observe that the integral term in the performance index of (3.2.1) may be written as

$$J_1 = \frac{1}{2} \int_{t_o}^{t_f} [x^T W_{uu} x + \lambda^T B W_{uu}^{-1} B^T \lambda] dt . \tag{3.3.4}$$

By defining

$$\chi = \begin{Bmatrix} x \\ \lambda \end{Bmatrix} , \quad \text{and } W = \begin{bmatrix} W_{xx} & 0 \\ 0 & B W_{uu}^{-1} B^T \end{bmatrix} , \tag{3.3.5}$$

one can rewrite J_1 as

$$J_1 = \frac{1}{2}\int_{t_o}^{t_f} \chi^T(t) W \chi(t) dt \ . \tag{3.3.6}$$

Using the state transition matrix solution of (3.3.1), we get

$$J_1 = \frac{1}{2}\chi^T(t_o)\int_{t_o}^{t_f} e^{G^T(t-t_o)} We^{G(t-t_o)} dt \chi(t_o) \ . \tag{3.3.7}$$

Solving for the matrix integral in (3.3.7), using the results or Appendix B, we obtain:

$$J_1 = \frac{1}{2}\chi^T(t_o) F_2^T(t_f-t_o) G_1(t_f-t_o)\chi(t_o) \ , \tag{3.3.8}$$

where

$$L = \begin{bmatrix} -G^T & W \\ 0 & G \end{bmatrix}, \quad \exp[L(t_f-t_o)] = \begin{bmatrix} F_1(t_f-t_o) & G_1(t_f-t_o) \\ 0 & F_2(t_f-t_o) \end{bmatrix},$$

$$F_1(t_f-t_o) = e^{-G^T(t_f-t_o)} \ ,$$

$$F_2(t_f-t_o) = e^{G(t_f-t_o)} \ ,$$

and

$$G_1(t_f-t_o) = \int_{t_o}^{t_f} e^{-G^T(t_f-t)} We^{G(t-t_o)} dt \ .$$

The matrix partitions F_1, F_2 and G_1 are computed by numerically exponentiating $L(t_f-t_o)$. For computing J_1 at fixed intervals of the final time, one can make use of the following propagation equations for F_1, F_2, and G_1:

$$F_i(t_f+\Delta t_f) = F_i(\Delta t_f)F_i(t_f) , \qquad i = 1, 2 ,$$

and

$$G_1(t_f+\Delta t_f) = F_1(\Delta t_f)G_1(t_f) + G_1(\Delta t_f)F_2(t_f) . \tag{3.3.9}$$

Using (3.2.1), (3.2.8), (3.3.3), (3.3.8), and (3.3.9), one can numerically compute values of J and Ω over a range of final times, and hence find the optimal t_f. For the case where one wishes to adjust W_t so that the optimal final time is at a desired value, one merely computes Ω using (3.2.8) and (3.3.3) at the desired final time and obtain the required value of W_t to make $\Omega = 0$.

Having found the values of the optimal final time and initial costates, the nominal state and control trajectories are given by

$$x^N(t) = [I\ 0]e^{G(t-t_o)} \begin{Bmatrix} x(t_o) \\ \lambda(t_o) \end{Bmatrix} ,$$

and

$$u^N(t) = -W_{uu}^{-1}B^T[0\ I]e^{G(t-t_o)} \begin{Bmatrix} x(t_o) \\ \lambda(t_o) \end{Bmatrix} . \tag{3.3.10}$$

3.4 Solution for the Feedback Gains

We now seek the solution for $\delta u(t)$ in the form

$$\delta u(t) = -K_1(t)\delta x(t) - K_2(t)d\psi , \tag{3.4.1}$$

where

$$u(t) = u^N(t) + \delta u(t) ,$$

$$\delta x(t) = x^N(t) - x(t) ,$$

K_1 and K_2 are the required feedback gains, and u(t) and x(t) are the perturbed controls and states. First, observing the symmetry of the coefficient matrices in (3.2.12) through (3.2.14), we re-write the equations in the form

$$\begin{Bmatrix} \delta\lambda_f \\ d\psi \\ d\Omega = 0 \end{Bmatrix} = \begin{bmatrix} S & R & m \\ R^T & Q & n \\ m^T & n^T & \alpha \end{bmatrix} \begin{Bmatrix} \delta x_f \\ d\nu \\ dt_f \end{Bmatrix}, \tag{3.4.2}$$

where

$$S = \left[\frac{\partial^2 \Phi}{\partial x^2}\right]\Big|_{t_f}, \quad R = \left[\frac{\partial \psi}{\partial x}\right]\Big|^T_{t_f}, \quad m = \left[\frac{\partial \Omega}{\partial x}\right]\Big|^T_{t_f},$$

and

$$Q = 0, \qquad n = \left[\frac{\partial \psi}{dt}\right]\Big|_{t_f}, \quad \alpha = \left[\frac{d\Omega}{dt}\right]\Big|_{t_f}.$$

To produce the desired form for the control correction, $\delta u(t)$, (3.4.2) is manipulated to the following form so that $\delta\lambda_f$ is a function of δx_f and $d\psi$ (see Appendix F):

$$\begin{Bmatrix} \delta\lambda_f \\ d\nu \\ dt_f \end{Bmatrix} = \begin{bmatrix} \hat{S} & \hat{R} & \hat{m} \\ \hat{R}^T & \hat{Q} & \hat{n} \\ \hat{m} & \hat{n} & \hat{\alpha} \end{bmatrix} \begin{Bmatrix} \delta x_f \\ -d\psi \\ -d\Omega = 0 \end{Bmatrix}, \tag{3.4.3}$$

where

$$\hat{S} = \bar{S} - \bar{R}\bar{\bar{Q}}^{-1}\bar{R}^T, \quad \hat{R} = -\bar{R}\bar{\bar{Q}}^{-1}, \quad \hat{m} = \bar{R}\bar{\bar{Q}}^{-1}\bar{n} - \bar{m},$$

$$\hat{Q} = -\bar{\bar{Q}}^{-1}, \qquad \hat{n} = \bar{\bar{Q}}^{-1}\bar{n}, \qquad \hat{\alpha} = -\bar{n}^T\bar{\bar{Q}}^{-1}\bar{n} - \bar{\alpha},$$

and

$$\bar{S} = S - \frac{mm^T}{\alpha}, \qquad \bar{R} = R - \frac{mn^T}{\alpha}, \qquad \bar{m} = \frac{m}{\alpha},$$

$$\bar{Q} = Q - \frac{nn^T}{\alpha}, \qquad \bar{n} = \frac{n}{\alpha}, \qquad \bar{\alpha} = \frac{1}{\alpha}. \tag{3.4.4}$$

Notice that $\bar{Q}$ is singular because $Q = 0$. Let us now assume that the coefficient matrix in (3.4.3) is a function of time, and that (3.4.3) is valid for values of the states and costates in $t\varepsilon[t_o,t_f]$, i.e.,

$$\begin{Bmatrix} \delta\lambda(t) \\ d\nu \\ dt_f \end{Bmatrix} = \begin{bmatrix} \hat{S}(t) & \hat{R}(t) & \hat{m}(t) \\ \hat{R}^T(t) & \hat{Q}(t) & \hat{n}(t) \\ \hat{m}^T(t) & \hat{n}^T(t) & \hat{\alpha}(t) \end{bmatrix} \begin{Bmatrix} \delta x(t) \\ -d\psi \\ -d\Omega = 0 \end{Bmatrix}. \tag{3.4.5}$$

The equations shown in (3.4.4) provide the terminal conditions for the variables $\hat{S}(t)$, $\hat{R}(t)$, $\hat{m}(t)$, $\hat{Q}(t)$, $\hat{n}(t)$, and $\hat{a}(t)$. Note that the variables S, R, m, Q, n, α, $\bar{S}$, $\bar{R}$, $\bar{m}$, $\bar{Q}$, $\bar{n}$, and $\bar{\alpha}$ still remain as constant matrices. The solutions for $\hat{Q}(t)$ and $\hat{\alpha}(t)$ are not needed for the control problem, but are carried out for completeness.

Treating $d\nu$, dt_f, $d\psi$, and $d\Omega$ as constants, one can differentiate (3.4.5) using (3.2.9) through (3.2.11) and collect terms to obtain the differential equations for the coefficient matrices:

$$\dot{\hat{S}} = -\hat{S}A - A^T\hat{S} + \hat{S}E\hat{S} - W_{xx}, \tag{3.4.6}$$

$$\dot{\hat{R}} = -[A - E\hat{S}]^T\hat{R}, \tag{3.4.7}$$

$$\dot{\hat{m}} = -[A - E\hat{S}]^T\hat{m}, \tag{3.4.8}$$

$$\dot{\hat{Q}} = \hat{R}^TE\hat{R}, \tag{3.4.9}$$

$$\dot{\hat{n}} = \hat{R}^TE\hat{m}, \tag{3.4.10}$$

$$\dot{\hat{\alpha}} = \hat{m}^T E \hat{m} , \tag{3.4.11}$$

where

$$E = B W_{uu}^{-1} B^T .$$

The solution of the Type 1 differential Riccati equation for $\hat{S}(t)$ is shown in Section 2.2.1 to be

$$\hat{S}(t) = \hat{S}_{ss} + Z^{-1}(t) , \tag{3.4.12}$$

where $\hat{S}_{ss}$ satisfies the steady-state Riccati equation

$$0 = -\hat{S}_{ss} A - A^T \hat{S}_{ss} + \hat{S} E \hat{S} - W_{xx} .$$

From Table 2-2, the solution for Z(t) can be shown to be

$$Z(t) = Z_{ss} + e^{\bar{A}(t-t_o)} [Z(t_o) - Z_{ss}] e^{\bar{A}^T(t-t_o)} , \tag{3.4.13}$$

where $\bar{A} = A - E\hat{S}_{ss}$ and Z_{ss} satisfies the steady-state Lyapunov equation for Z(t) defined following (2.2.10). Since $\hat{R}(t)$ and $\hat{m}(t)$ satisfy differential equations of Type 2, their solutions are given by (see Tables 2-1 and 2-2):

$$\hat{R}(t) = Z^{-1}(t) e^{\bar{A}(t-t_o)} Z(t_o) \hat{R}(t_o) , \tag{3.4.14}$$

and

$$\hat{m}(t) = Z^{-1}(t) e^{\bar{A}(t-t_o)} Z(t_o) \hat{m}(t_o) . \tag{3.4.15}$$

Since $\hat{Q}(t)$, $\hat{n}(t)$, and $\hat{\alpha}(t)$ satisfy differential equations of Type 4, their solutions are obtained from Table 2-1 as

$$\hat{Q}(t) = \hat{R}^T(t)Z(t)\hat{R}(t) + \hat{Q}(t_o) - \hat{R}^T(t_o)Z(t_o)\hat{R}(t_o) , \qquad (3.4.16)$$

$$\hat{n}(t) = \hat{R}^T(t)Z(t)\hat{m}(t) + \hat{n}(t_o) - \hat{R}^T(t_o)Z(t_o)\hat{m}(t_o) , \qquad (3.4.17)$$

and

$$\hat{\alpha}(t) = \hat{m}^T(t)Z(t)\hat{m}(t) + \hat{\alpha}(t_o) - \hat{m}^T(t_o)Z(t_o)\hat{m}(t_o) . \qquad (3.4.18)$$

The boundary conditions for these coefficient matrices are given by expresions shown in (3.4.4); however, since Q is singular, the terminal conditions of(3.4.4) must be converted to equivalent initial conditions for numerical computation. For the $\hat{S}(t)$ equation, the terminal condition for Z(t) is obtained from (3.4.4) and (3.4.12):

$$Z(t_f) = [\tilde{S} - \bar{R}\bar{Q}^{-1}\bar{R}^T]^{-1} , \qquad (3.4.19)$$

where

$$\tilde{S} = \bar{S} - \hat{S}_{ss} .$$

Equation (3.4.19) is re-written in a computable form using the matrix inversion lemma:

$$Z(t_f) = \tilde{S}^{-1} - \tilde{S}^{-1}\bar{R}[\bar{R}^T\tilde{S}^{-1}\bar{R} - \bar{Q}]^{-1}\bar{R}^T\tilde{S}^{-1} . \qquad (3.4.20)$$

For nondegenerate cases, it can be shown that $\tilde{S}$ and the term in brackets in (3.4.20) are non-singular. The inverse of $Z(t_f)$, however, does not exist. The initial condition for Z(t) is given by (3.4.13) as

$$Z(t_o) = Z_{ss} + e^{-\bar{A}(t_f-t_o)}[Z(t_f) - Z_{ss}]e^{-\bar{A}^T(t_f-t_o)} . \qquad (3.4.21)$$

Repeated application of the matrix inversion lemma leads to the following

expressions for the initial conditions for $\hat{R}(t)$, $\hat{m}(t)$, $\hat{Q}(t)$, $\hat{n}(t)$, and $\hat{\alpha}(t)$ (see Appendix D):

$$\hat{R}(t_o) = Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}\bar{S}^{-1}\bar{R}[\bar{R}^T\bar{S}^{-1}\bar{R} - \bar{Q}]^{-1} , \tag{3.4.22}$$

$$\hat{m}(t_o) = - Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}[Z(t_f)\bar{m} + \bar{S}^{-1}\bar{R}[\bar{R}^T\bar{S}^{-1}\bar{R} - \bar{Q}]^{-1}\bar{n}] , \tag{3.4.23}$$

$$\hat{Q}(t_o) = \hat{R}^T(t_o)Z(t_o)\hat{R}(t_o) + [\bar{R}^T\bar{S}^{-1}\bar{R} - \bar{Q}]^{-1} , \tag{3.4.24}$$

$$\hat{n}(t_o) = \hat{R}^T(t_o)Z(t_o)\hat{m}(t_o) + [\bar{R}^T\bar{S}^{-1}\bar{R} - \bar{Q}]^{-1}(\bar{R}^T\bar{S}^{-1}\bar{m} - \bar{n}) , \tag{3.4.25}$$

and

$$\hat{\alpha}(t_o) = [\bar{n}^T\hat{R}^T(t_o) + \hat{m}^T(t_o) + \bar{m}^T e^{\bar{A}(t_f-t_o)}]Z(t_o)\hat{m}(t_o) - \bar{n}^T\hat{n}(t_o) - \bar{\alpha} \tag{3.4.26}$$

Using (3.2.11) and (3.4.5), one can write the feedback gains of (3.4.1) as

$$K_1(t) = W_{uu}^{-1}B^T\hat{S}(t) , \tag{3.4.27}$$

and

$$K_2(t) = - W_{uu}^{-1}B^T\hat{R}(t) , \tag{3.4.28}$$

where the solutions of $\hat{S}(t)$ and $\hat{R}(t)$ are given by (3.4.12) and (3.4.14), respectively. Appendix G shows the propagation scheme for efficiently computing $\hat{S}(t)$, $\hat{R}(t)$, $\hat{m}(t)$, and $\hat{n}(t)$ at discrete points in time. (The variables $\hat{m}(t)$ and $\hat{n}(t)$ are used for estimating the change in the final time, as discussed in the following section.)

3.5 Time-To-Go Indexing Scheme

On observing the time arguments in (3.4.1) and recalling the fact that the final time is free, it quickly becomes apparent that if the optimal final time of the perturbed system lies beyond the nominal final time, then the feedback gains are undefined for part of the time ($t > t_f^N$) along the neighboring extremal path. One of the methods suggested for eliminating this problem is the use of time-to-go indexing [6,57,85] so that the time-to-go on the perturbed trajectory is the same as the time-to-go on the nominal trajectory (see Figure 3-1). Equation (3.4.1) is then re-written as

$$\delta u(t) = - K_1(t_I)\delta x(t) - K_2(t_I)d\psi \ , \qquad (3.5.1)$$

where

$$t_f^N - t_I = t_f - t \ ,$$

$$dt_f = t - t_I = t_f - t_f^N \ ,$$

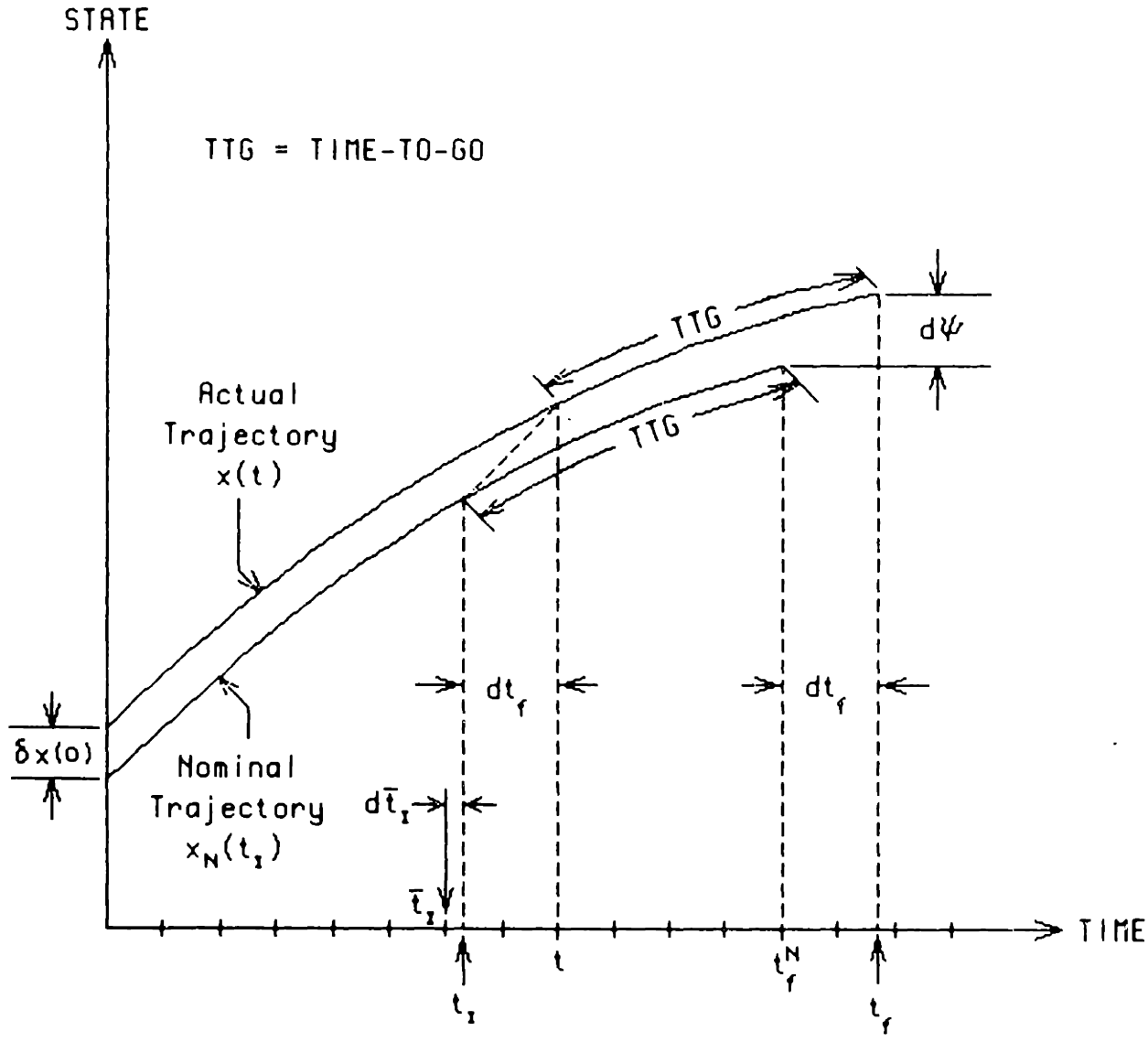
t is the current time, t_I is the indexed time, t_f^N is the nominal final time, and t_f is the final time on the perturbed trajectory. To compute the indexed time, let us re-write the last row of (3.4.5) for the change in final time:

$$dt_f = \hat{m}^T(t_I)\delta x(t) - \hat{n}^T(t_I)d\psi \ , \qquad (3.5.2)$$

where the time arguments for $\hat{m}^T(t)$ and $\hat{n}^T(t)$ have been changed to t_I. Since the feedback gains $\hat{S}(t)$, $\hat{R}(t)$, $\hat{m}(t)$, and $\hat{n}(t)$ are efficiently propagated at fixed time intervals, let us define $\bar{t}_I$ as the points in time at which the values of the feedback gains are available:

$$\bar{t}_I = n\Delta t, \ n = 1,2,3, \ \ldots \qquad (3.5.3)$$

where Δt is the propagation time step.



t = Current Time t_I = Indexed Time

t_f^N = Nominal Final Time t_f = Actual Final Time

$\bar{t}_I$ = Time-points at which Gains are Computed

Figure 3-1. Time-To-Go Indexing Scheme

In (3.5.2), the definition of $d\psi$ is

$$d\psi = \psi[x(t_f),t_f] - \psi[x^N(t_f^N),t_f^N] , \tag{3.5.4}$$

where $x^N(t)$ is the nominal state trajectory. Note that the second term of (3.5.4) is identically equal to zero. Let us assume that the perturbed terminal manifold is given by

$$Mx(t_f) = \psi_d(t_f) . \tag{3.5.5}$$

Substituting (3.5.5) into (3.2.3) leads to

$$d\psi = \psi_d(t_f) - \psi_d^N(t_f) . \tag{3.5.6}$$

3.5.1 Local Quadratic Fit for the Feedback Gains

In order to compute the gains $\hat{m}(t)$ and $\hat{n}(t)$ at the indexed time, let us perform a quadratic fit of those gains over three discrete points in time -- $\bar{t}_I-\Delta t$, $\bar{t}_I$, and $\bar{t}_I+\Delta t$ -- and interpolate to the indexed time t_I. For a quadratic fit of a variable V(t) (whether a scalar, vector, or matrix), let us assume that we have three values of V(t) at different points in time, i.e., $V(t_1)$, $V(t_2)$, and $V(t_3)$, and assume that the variation in V(t) can be modelled as being locally quadratic, i.e.,

$$V(t) = Ft^2 + Gt + H . \tag{3.5.7}$$

Writing (3.5.7) for each of the three time points, and for each element of V(t) (assuming V(t) a matrix for now), we obtain

$$\begin{bmatrix} t_1^2 & t_1 & 1 \\ t_2^2 & t_2 & 1 \\ t_3^2 & t_3 & 1 \end{bmatrix} \begin{Bmatrix} F_{ij} \\ G_{ij} \\ H_{ij} \end{Bmatrix} = \begin{Bmatrix} V_{ij}(t_1) \\ V_{ij}(t_2) \\ V_{ij}(t_3) \end{Bmatrix} . \qquad (3.5.8)$$

The elements of the coefficient matrices F, G, and H may be computed from (3.5.8) by solving for F_{ij}, G_{ij}, and H_{ij}, yielding

$$\begin{Bmatrix} F_{ij} \\ G_{ij} \\ H_{ij} \end{Bmatrix} = \frac{1}{D} \begin{bmatrix} t_2-t_3 & t_3-t_1 & t_1-t_2 \\ t_3^2-t_2^2 & t_1^2-t_3^2 & t_2^2-t_1^2 \\ t_2^2t_3-t_3^2t_2 & t_3^2t_1-t_1^2t_3 & t_1^2t_2-t_2^2t_1 \end{bmatrix} \begin{Bmatrix} V_{ij}(t_1) \\ V_{ij}(t_2) \\ V_{ij}(t_3) \end{Bmatrix} , \qquad (3.5.9)$$

where

$$D = t_1^2t_2 - t_1^2t_3 + t_3^2t_1 - t_2^2t_1 + t_2^2t_3 - t_3^2t_2 .$$

When using three values of gains at times $\bar{t}_I-\Delta t$, $\bar{t}_I$, and $\bar{t}_I+\Delta t$, one can define the time variables of (3.5.9) so that $t_2=0$ corresponds to $\bar{t}_I$. Thus, for the other time values, $t_1=-\Delta t$ and $t_3=\Delta t$ correspond to the times $\bar{t}_I-\Delta t$ and $\bar{t}_I+\Delta t$, respectively. Using these values for t_1, t_2, and t_3, we can re-write (3.5.9) as

$$\begin{Bmatrix} F_{ij} \\ G_{ij} \\ H_{ij} \end{Bmatrix} = \frac{-1}{2(\Delta t)^3} \begin{bmatrix} -\Delta t & 2\Delta t & -\Delta t \\ (\Delta t)^2 & 0 & -(\Delta t)^2 \\ 0 & -2(\Delta t)^3 & 0 \end{bmatrix} \begin{Bmatrix} V_{ij}(\bar{t}_I-\Delta t) \\ V_{ij}(\bar{t}_I) \\ V_{ij}(\bar{t}_I+\Delta t) \end{Bmatrix} . \qquad (3.5.10)$$

To interpolate the gains to the indexed time, we define

$$d\bar{t}_I = t_I - \bar{t}_I \ , \tag{3.5.11}$$

where $\bar{t}_I$ is a discrete time-point which is close to t_I, so that $d\bar{t}_I$ is smaller than Δt in magnitude. Using (3.5.7), with time defined relative to $\bar{t}_I$, we get the interpolation

$$V_{ij}(t_I) = F_{ij}(d\bar{t}_I)^2 + G_{ij}(d\bar{t}_I) + H_{ij} \ . \tag{3.5.12}$$

Substitution of (3.5.10) into (3.5.12) yields

$$V_{ij}(t_I) = \frac{1}{2}(d^2-d)V_{ij}(\bar{t}_I-\Delta t) + (1-d^2)V_{ij}(\bar{t}_I)$$

$$+ \frac{1}{2}(d^2+d)V_{ij}(\bar{t}_I+\Delta t) \ , \tag{3.5.13}$$

where

$$d = d\bar{t}_I/\Delta t \ .$$

3.5.2 Solution for the Indexed Time

The indexed time is then found by the interpolation of (3.5.13) for $\hat{m}(t_I)$ and $\hat{n}(t_I)$, and an iteration involving (3.5.2). Substitution of (3.5.11) into (3.5.2) gives

$$t - \bar{t}_I - d\bar{t}_I = \hat{m}^T(t_I)\delta x(t) - \hat{n}^T(t_I)d\psi \ , \tag{3.5.14}$$

where we have used the fact that $dt_f = t-t_I$. If the value of $d\bar{t}_I$ is correct, (3.5.14) would be satisfied. However, since $d\bar{t}_I$ cannot be computed explicitly, it must be computed iteratively. The error in satisfying (3.5.14) may be defined as

$$\varepsilon = \hat{m}^T(t_I)\delta x(t) - \hat{n}^T(t_I)d\psi - t + \bar{t}_I + d\bar{t}_I \ . \qquad (3.5.15)$$

One way of solving for the correct indexed time is to iteratively substitute values of $d\bar{t}_I$ into (3.5.15) and (3.5.13) for $\hat{m}(t_I)$ and $\hat{n}(t_I)$ [85], with the corrections

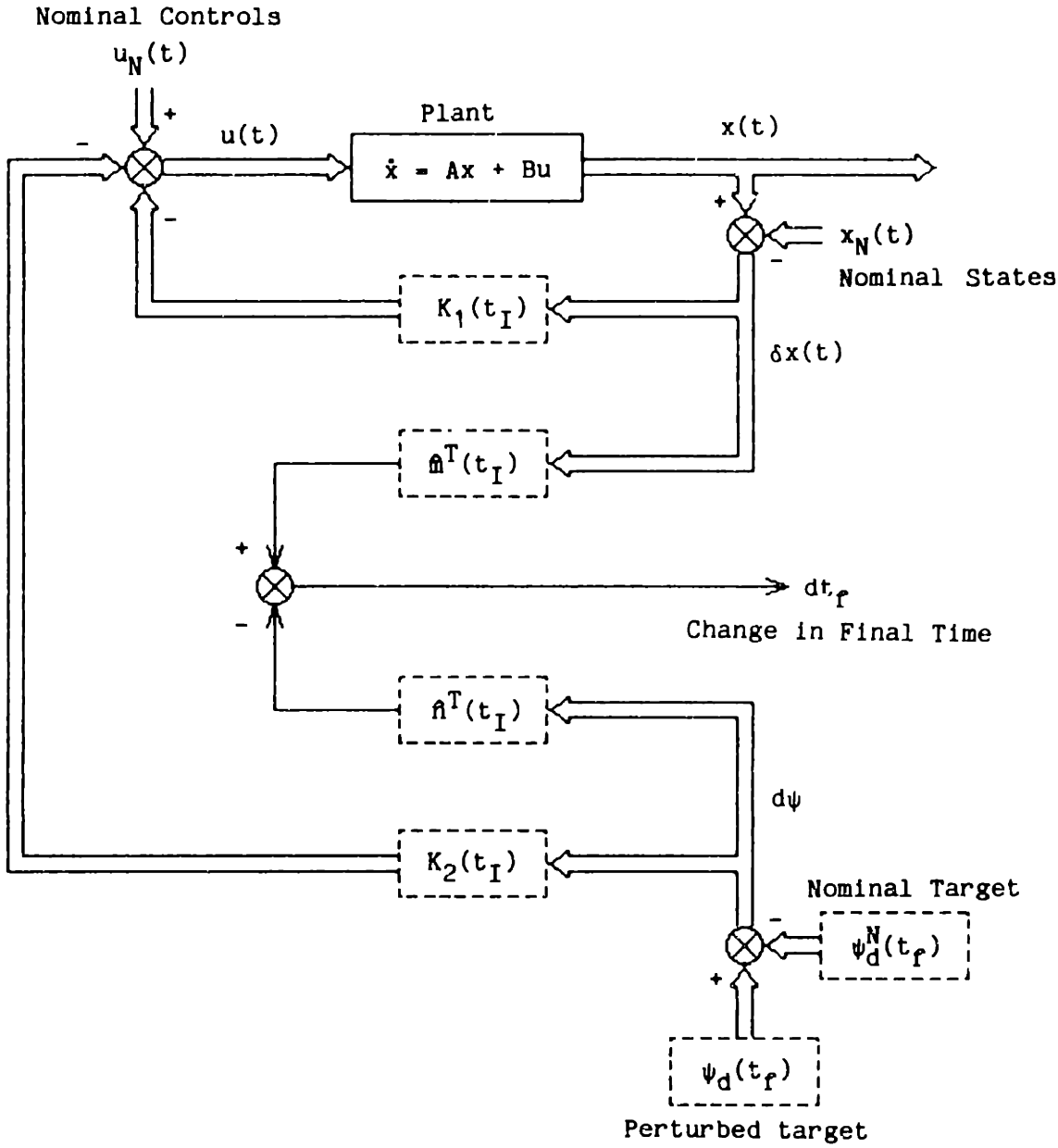
$$d\bar{t}_I := d\bar{t}_I - \varepsilon \qquad (3.5.16)$$

performed after each substitution. This Jacobi or Gauss-Siedel type of iteration [34] fails if $\hat{m}(t)$ or $\hat{n}(t)$ varies rapidly. When this occurs, one may successively decrease the amount of correction applied in (3.5.16) by halves or temporarily decrease the propagation step, Δt, until convergence is achieved.

The logic for propagating the gains is as follows. If $d\bar{t}_I > \Delta t$, then $\bar{t}_I$ is incremented by Δt, and the gains are propagated forward by one time-step. If $d\bar{t}_I < -\Delta t$, then $\bar{t}_I$ is decremented by Δt, and the gains are propagated backwards by one time-step. The above propagation is repeated, if necessary, until $|d\bar{t}_I| < \Delta t$. When $\bar{t}_I$ is incremented until $\bar{t}_I = t_f^N$, then the end of the maneuver is reached. Backward propagation of the gains, though not occuring often, may be needed when there are disturbances acting on the system, or when there is a sudden change in the terminal manifold during the maneuver. Figure 3-2 shows the block diagram for the free-final-time perturbation feedback controller.

3.6 Illustrative Examples

The specific model considered in this section consists of a rigid hub with four identical elastic appendages attached symmetrically about the central hub, and is derived from the experimental structure of [13], using NASTRAN data (see Fig. 3-3). In particular, the following idealizations are considered: (i) single-axis maneuvers, (ii) in-plane motion, (iii) antisymmetric deformations, (iv) small linear flexural deformations, (v) only the linear time-invariant form of the equations of motion are considered, and (vi) the control actuator is modeled as a



Dashed lines indicate variables which depend on dt_f, the change in final time.

Figure 3-2. Block Diagram for the Free-Final-Time Perturbation Feedback Controller

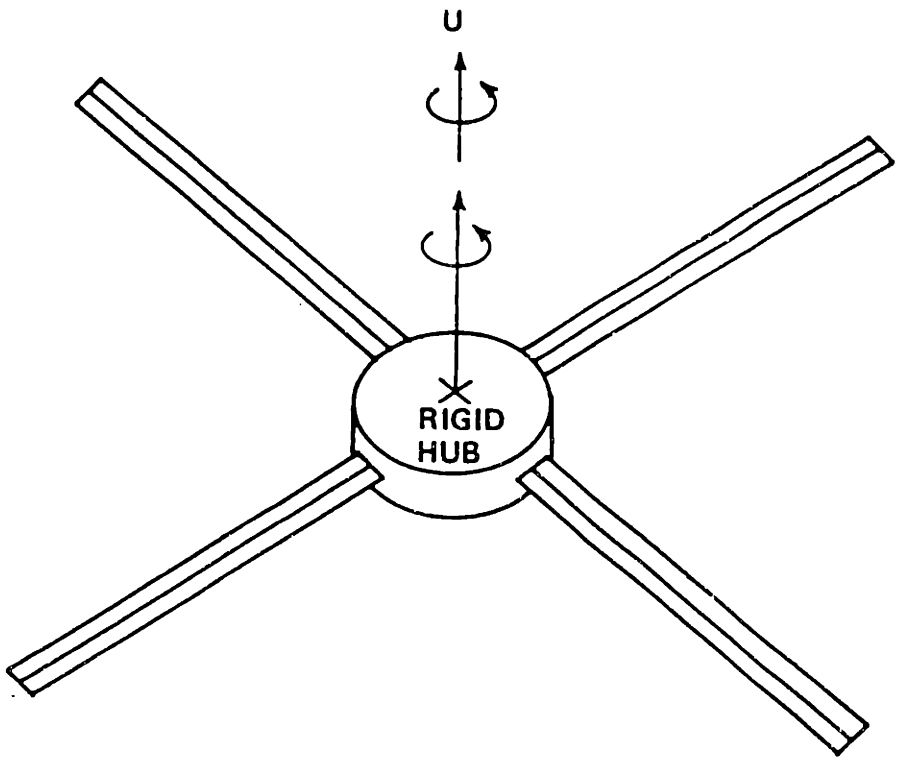


Figure 3-3. Model Structure

concentrated torque generating device. Figure 3-4 shows the first three antisymmetric modes, which, with the rigid body mode, defines the full-order model. The control system for the vehicle consists of a single controller in the rigid part of the structure. The structural parameters of the model are presented in [13]. Because of the above assumptions, only the antisymmetric modes are used for the example cases in this section. In addition, full state feedback is assumed.

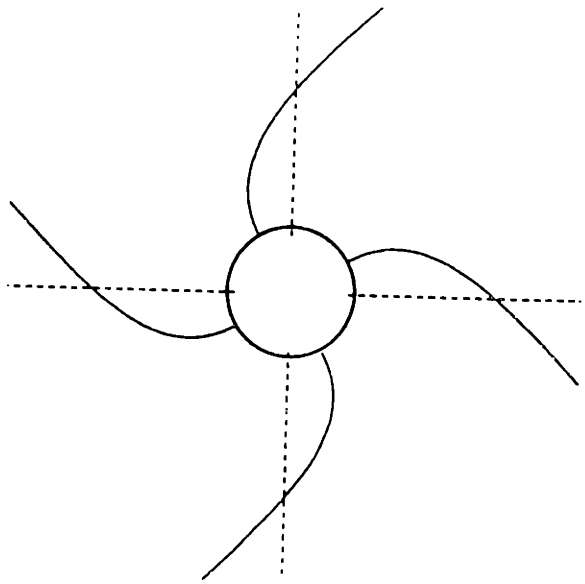
The rigid body mode and the first elastic mode are chosen for inclusion in the state vector for the control problem. Hence, the second and third elastic modes represent residual modes. Control smoothing is done by penalizing the first and second time derivatives of the control in the performance index, and augmenting the state vector with the control and its first time derivative (Appendix C). The state vector is given by

$$x = [\eta_o \ \eta_1 \ \dot{\eta}_o \ \dot{\eta}_1 \ u \ \dot{u}]^T , \qquad (3.6.1)$$

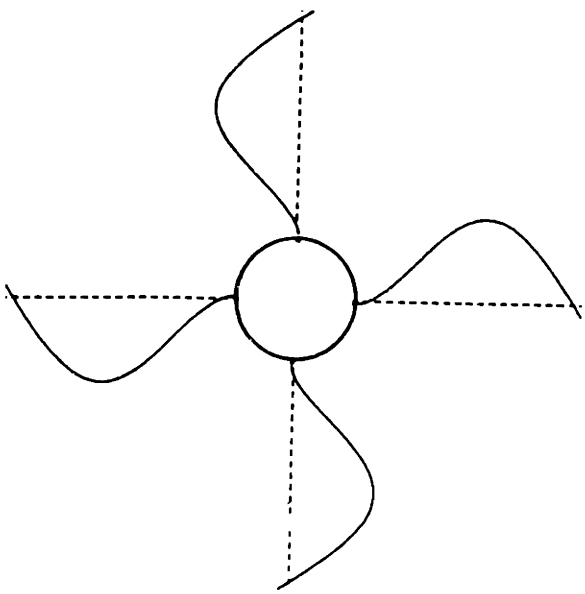
where η_o and η_1 are the amplitudes of the rigid mode and first elastic mode, and u is the control torque.

As a result of many numerical simulations, it has been found that without modifications, the optimal perturbation feedback control, as presented in this chapter, performs poorly, especially towards the final time, when the gains are large and vary quickly. One source of difficulty is due to the singularity of $\bar{Q}$. Since $\bar{Q}$ is singular, all of the feedback gains become infinite at the final time. Because of numerical inaccuracies, this results in randomly large gains near the final time. A remedy is to use $Q = -\varepsilon_1 I$, where ε_1 is a small positive number. The negative sign for Q is due to the fact that if (3.4.2) were to hold for all $t\varepsilon[t_o,t_f]$, it can be shown that $Q < 0$ for $t < t_f$. With this modification, the gains become large in a well-behaved manner near the final time.

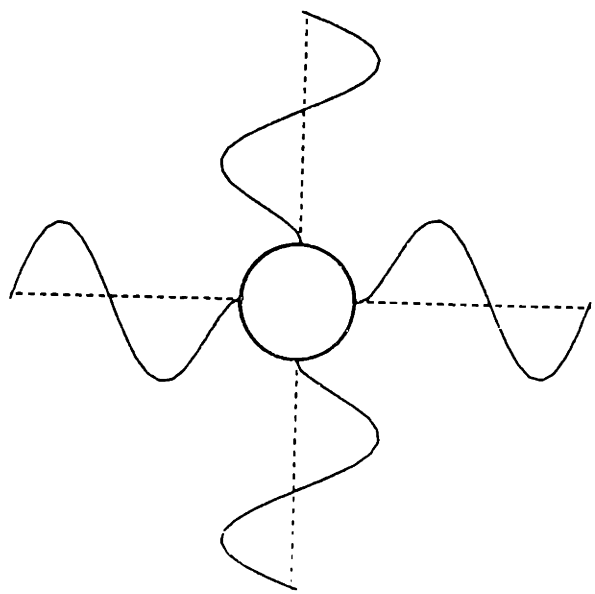
Another difficulty manifests itself in the numerical instability of the final time estimation. Since the correction variable, ε, of (3.5.15)



MODE 1, 1.261 Hz



MODE 2, 8.396 Hz



MODE 3, 24.905 Hz

Figure 3-4. Elastic Deflection Modes

is computed as the difference of potentially large numbers (relative to ε), the calculation is easily numerically unstable when the values of $\hat{m}(t_I)$ and $\hat{n}(t_I)$ are large or corrupted by numerical inaccuracy. This results in values of ε alternating in sign and increasing in magnitude at each successive iteration; that is, the iterative algorithm becomes unstable. A remedy for this problem is to increase the magnitude of α in (3.4.2), using

$$\alpha = (1 + \varepsilon_2) \left[\frac{d\Omega}{dt}\right]\Big|_{t_f} , \tag{3.6.2}$$

where ε_2 is a small positive number. It is found that on choosing $\varepsilon_2 = 0.02$, the final time estimates become much better behaved, and the errors in satisfaction of the terminal constraints are reduced by about two or three orders of magnitude. If we use $\varepsilon_2 = 0.5$, the terminal errors are reduced by almost four orders of magnitude. However, with $\varepsilon_2 = 0.5$ the free-final-time controller behaves as if it were a fixed-final-time controller. From these observations, it is clear that the performance of the controller is extremely sensitive to the value of α. It is not obvious, however, that α should be modified rather than any other variable. Nor is it obvious what the mechanism is for the improvement of system performance, since the modification of α affects all of the gain variables. However, it is interesting to note that the modified variables, Q and α, are diagonal blocks of the matrix coefficient of (3.4.2). The remaining diagonal block, $S = S_f$, also greatly affects system performance, as discussed in [19]. For the results of this section, values chosen for ε_1 and ε_2 are 10^{-10} and 0.02, respectively.

Cases 1 and 2 represent spin-up maneuvers where the final hub angle is free, while the hub angular rate is specified via the terminal constraint function ψ:

$$\psi[x(t_f),t_f] = \dot{\theta}[x(t_f)] - 0.05 = 0 , \tag{3.6.3}$$

where $\dot{\theta}$ represents the angular rate of the rigid hub, and is a linear function of $\dot{\eta}_o$ and $\dot{\eta}$. The weighting matrices for Cases 1 and 2 are given

in Table 3-1. The weight on the elapsed time is adjusted so that the nominal final time is 5.0 s, for convenience. The terminal state weight is computed using the algorithm of [19], modified to include terminal constraints. Note that the columns and rows corresponding to the first and third elements of the final state vector have relatively small numbers. This is because the hub angle is free and the hub angular velocity is specified via the terminal constraints. (Theoretically, one could set those rows and columns to zero; however, doing so would lead to the difference $S_f - \hat{S}_{ss}$ becoming indefinite. This leads to an indefinite Z(t) matrix with a large condition number, which is numerically undesirable.) The large values of the remaining diagonal elements in S_f indicate that the terminal values of η_1, $\dot{\eta}_1$, u, and $\dot{u}$ are required to be small. The state weight W_{xx} is specified using diagonal weights for θ, η_1, $\dot{\theta}$, $\dot{\eta}_1$, u, and $\dot{u}$. The off-diagonal terms of W_{11} and W_{22} in Table 3-1 result from transforming the weights for θ, η_1, $\dot{\theta}$, and $\dot{\eta}_1$ into weights for η_0, η_1, $\dot{\eta}_0$, and $\dot{\eta}_1$. The perturbed initial conditions for Cases 1 and 2 are the same. For Case 2, the terminal constraint is the nominal one from 0 to 2.5 s. At 2.5 s, the value of the new terminal constraint is entered into the feedback calculation, replacing the nominal value. This represents a sudden change in the desired final hub angular rate.

The modal space amplitudes and rates are transformed into the physical quantities in time history plots for Case 1 (Fig. 3-5) and Case 2 (Fig. 3-6). The dotted curves represent the nominal trajectories, while the solid curves represent the perturbed trajectories. The short dotted lines in the angular velocity plots represent the target velocities for the nominal and perturbed trajectories. For Case 1, the hub angular velocity achieves the perturbed final value. The final values of the mid-span relative angular rate, tip displacement, control, and first time derivative of the control are very close to zero as desired. For Case 2, the hub angular velocity also achieves the perturbed final value. The final values of the mid-span relative angular rate, tip displacement, control, and control rate are also small as desired. Also, notice the jump discontinuity in the second time derivative of the control torque. This is due to the jump discontinuity in $d\psi$ at $t = 2.5$ s. The effect on the state trajectory of the jump discontinuity in the second time

Table 3-1. Maneuver Specifications for Cases 1 and 2

Weights

$$S_f = \begin{bmatrix} 2.63 & & & & & \text{Symmetric} \\ 3.95 & 3.08(8) & & & & \\ 7.90(-3) & -7.31(-1) & 2.63 & & & \\ 1.18(-2) & -4.01(6) & 3.95 & 2.62(7) & & \\ 6.22(-6) & -3.27(3) & 4.16(-3) & 1.91(4) & 1.00(5) & \\ 5.03(-9) & -3.78 & 4.68(-6) & 2.15(1) & 1.88(2) & 2.50(3) \end{bmatrix}$$

W_{xx} = Block diagonal [W_{11}, W_{22}, 1.00(-5), 1.00(-5)]

$$W_{11} = \begin{bmatrix} 2.63(-7) & 3.95(-7) \\ 3.95(-7) & 1.06(-3) \end{bmatrix} \qquad W_{22} = \begin{bmatrix} 2.63(-5) & 3.95(-5) \\ 3.95(-5) & 1.06(-3) \end{bmatrix}$$

W_{uu} = [10]

Nominal conditions:

$$\theta(t_o) = \dot{\theta}(t_o) = \eta_1(t_o) = \dot{\eta}_1(t_o) = u(t_o) = \dot{u}(t_o) = 0$$

$$\psi = \dot{\theta}(t_f) - 0.05 = 0$$

Perturbed Conditions:

$$\theta(t_o) = 0.01 \text{ rad}, \quad \dot{\theta}(t_o) = -0.005 \text{ rad/s}$$

$$\eta_1(t_o) = \dot{\eta}_1(t_o) = u(t_o) = \dot{u}(t_o) = 0$$

$$\dot{\theta}(t_f) = 0.055$$

a(n) indicates $a \times 10^n$

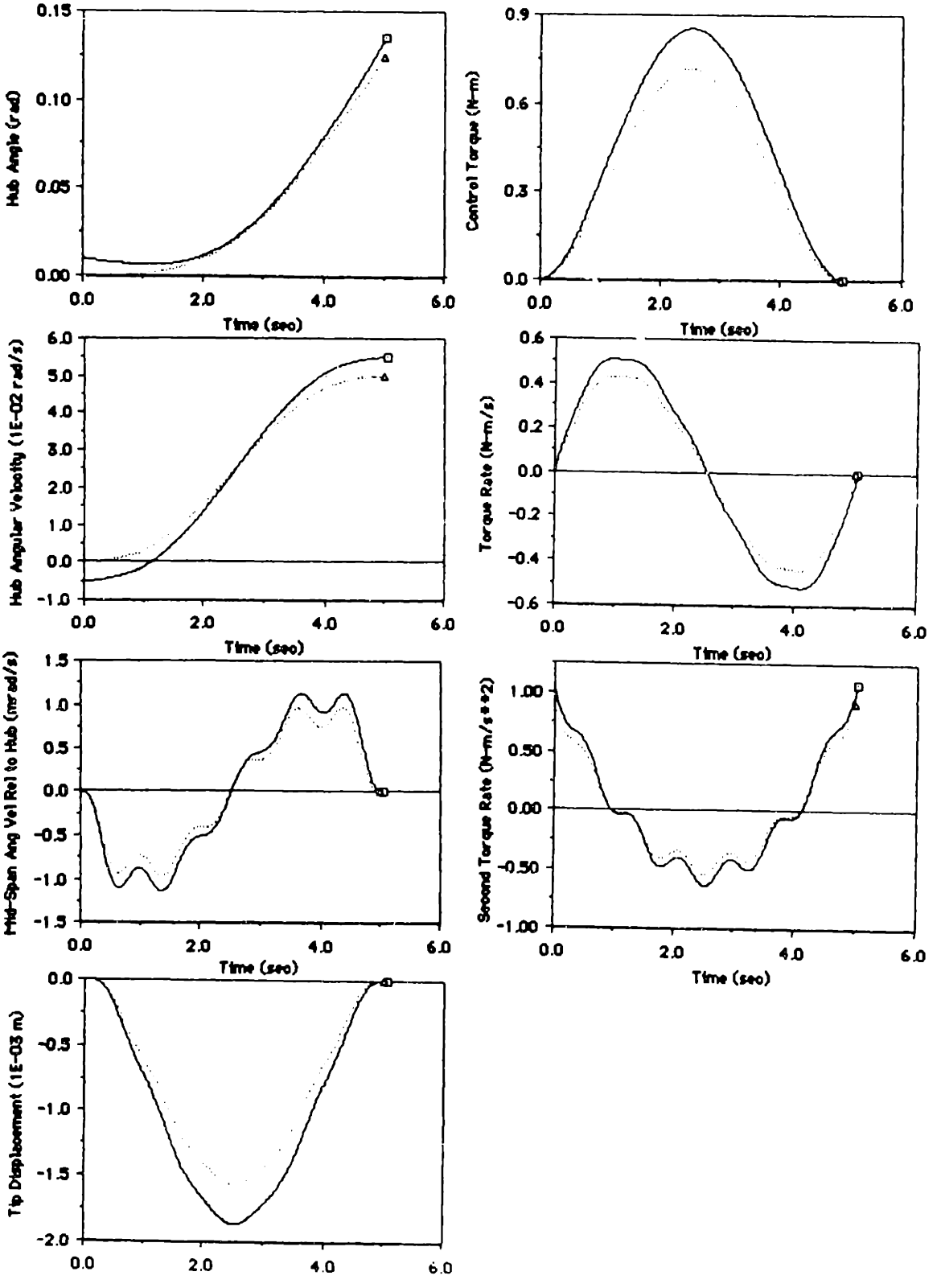


Figure 3-5. Case 1. Spin-Up Maneuver

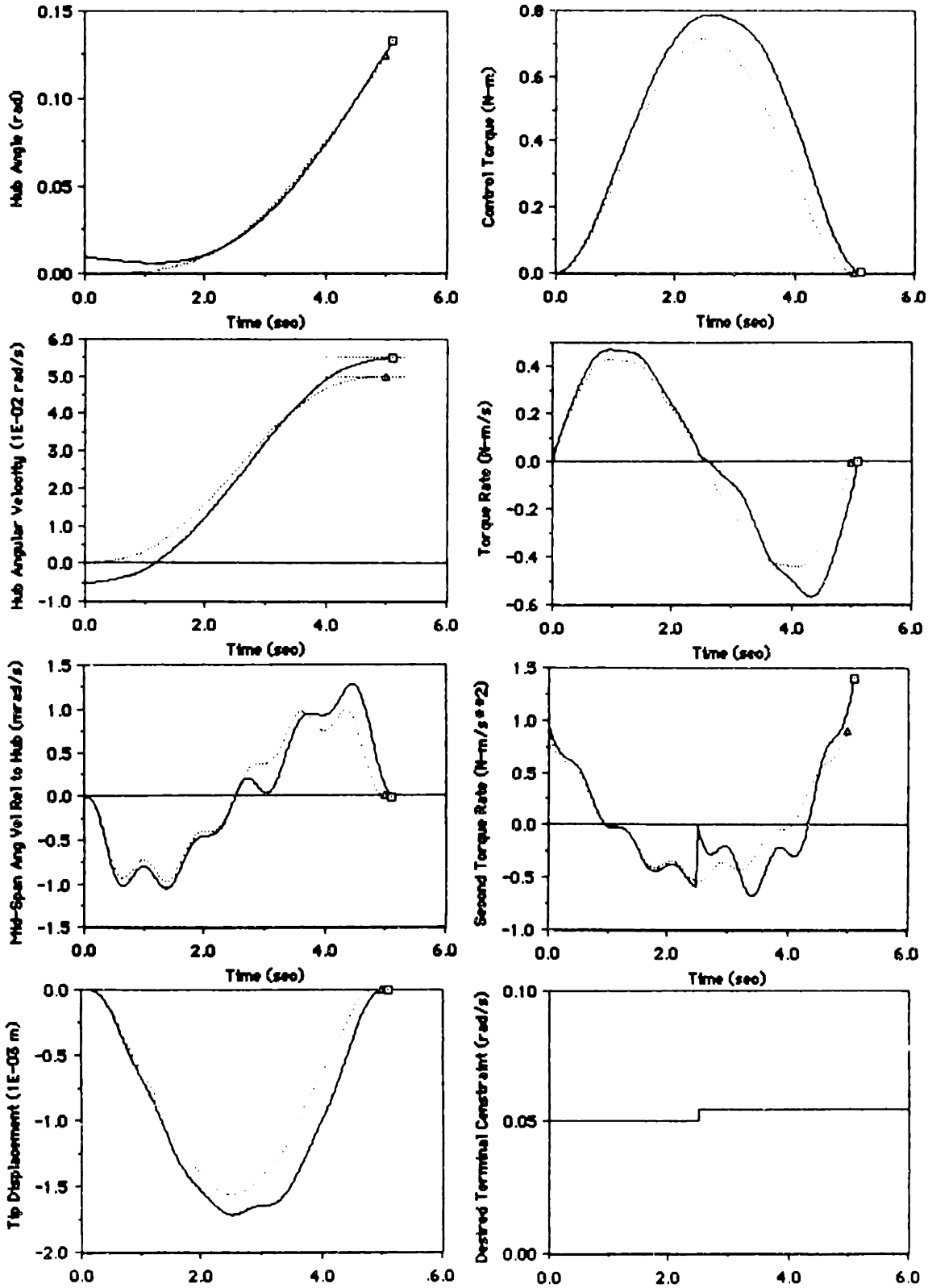


Figure 3-6. Case 2. Spin-Up Maneuver with Changing Terminal Constraints

derivative of the control torque is minimized by the use of control smoothing.

Cases 3 and 4 represent retargeting maneuvers, where the final hub orientation, angular rate, angular acceleration, and third time derivative of the hub angle are required to match a moving target whose motion is presumably known. The nominal target motion is a linear fly-by (see Fig. 3-7), where the target travels in a straight line at constant velocity. The structure is assumed to rotate about the z axis, with the appendages moving in the x-y plane. The components of the ψ vector are given in the following equations:

$$\theta(t_f) - \theta_T(t_f) = 0, \quad \theta_T(t_f) = \tan^{-1}\left(\frac{y}{x}\right),$$

$$\dot{\theta}(t_f) - \dot{\theta}_T(t_f) = 0, \quad \dot{\theta}_T(t_f) = \frac{vx}{x^2 + y^2},$$

$$\ddot{\theta}(t_f) - \ddot{\theta}_T(t_f) = 0, \quad \ddot{\theta}_T(t_f) = \frac{-2v^2xy}{(x^2 + y^2)^2},$$

and

$$\dddot{\theta}(t_f) - \dddot{\theta}_T(t_f) = 0, \quad \dddot{\theta}_T(t_f) = \frac{8v^3xy^2}{(x^2 + y^2)^3} - \frac{2v^3x}{(x^2 + y^2)^2}, \tag{3.6.4}$$

where $x = x_o$ and $y = y_o + vt_f$. The perturbed target motion is also a constant-velocity linear fly-by, but with a different starting location and different velocity. The target parameters and weighting matrices are shown in Table 3-2. As for Cases 1 and 2, the weight on the elapsed time is adjusted so that the nominal final time is 5.0 s, for convenience. The terminal state weighting is also computed using a modified version of the algorithm in [19]. Because of the large terminal weights on η_1 and $\dot{\eta}_1$, the final values of the remaining state variables are more or less constrained via the terminal constraint conditions of (3.6.4). Therefore, the elements in the rows and columns of S_f corresponding to η_o, $\dot{\eta}_o$, u, and $\dot{u}$ can be set to very small numbers. However, because of the numerical

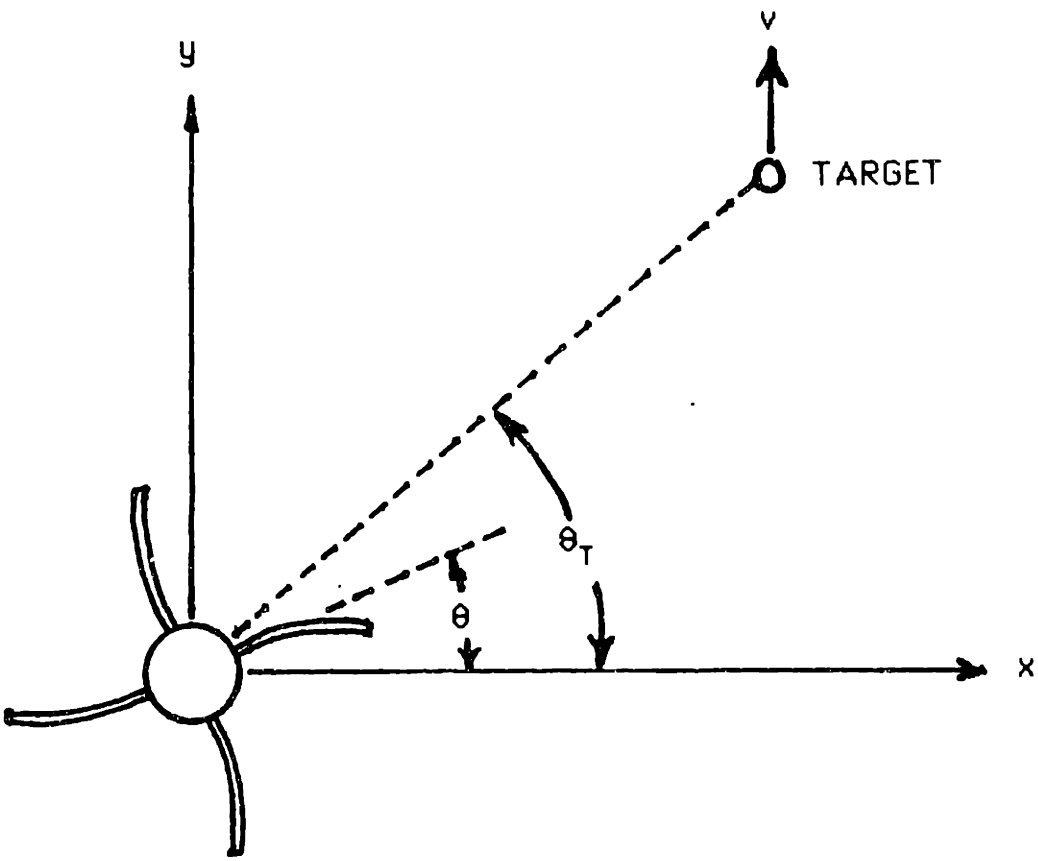


Figure 3-7. Linear Fly-By for Cases 3 and 4

Table 3-2. Maneuver Specifications for Cases 3 and 4

Weights

$$S_f = \begin{bmatrix} 2.63(3) & & & & & \text{Symmetric} \\ 3.95(3) & 3.09(6) & & & & \\ 7.90 & -7.31(2) & 2.63(3) & & & \\ 1.18(1) & 4.12(4) & 3.95(3) & 2.68(5) & & \\ 6.24(-3) & -3.38(1) & 4.16 & 1.98(2) & 1.00(5) & \\ 4.73(-6) & -3.52(-2) & 4.23(-3) & 2.01(-1) & 1.55(2) & 3.23(3) \end{bmatrix}$$

W_{xx} = Block diagonal [W_{11}, W_{22}, 1.00(-5), 1.00(-5)]

$$W_{11} = \begin{bmatrix} 2.63(-5) & 3.95(-5) \\ 3.95(-5) & 6.91(-5) \end{bmatrix} \qquad W_{22} = \begin{bmatrix} 2.63(-5) & 3.95(-5) \\ 3.95(-5) & 6.91(-5) \end{bmatrix}$$

W_{uu} = [10]

Nominal conditions:

$\theta(t_o) = \dot{\theta}(t_o) = \eta_1(t_o) = \dot{\eta}_1(t_o) = u(t_o) = \dot{u}(t_o) = 0$

$x_o = 2.7(7)$, $y_o = -4.0(4)$, $v = 8.0(3)$

Perturbed Conditions:

$\theta(t_o) = 2(-5)$rad , $\dot{\theta}(t_o) = -2(-5)$ rad/s

$\eta_1(t_o) = \dot{\eta}_1(t_o) = u(t_o) = \dot{u}(t_o) = 0$

$x_o = 2.5(7)$, $y_o = -4.1(4)$, $v = 7.8(3)$

a(n) indicates $a \times 10^n$

considerations mentioned in the discussion of Cases 1 and 2, these elements are set to larger numbers to decrease the condition number of the Z(t) matrix. The elements of W_{xx} and W_{uu} are obtained as for Cases 1 and 2. As in Case 2, the perturbed trajectory information for Case 4 is introduced into the feedback law at t = 2.5 s. This simulates the case when target information is updated during a maneuver.

The time history plots for Cases 3 and 4 are shown in Figures 3-8 through 3-10. For Case 3, note that all the terminal constraints are satisfied for both the nominal and perturbed trajectories. (The target angular acceleration, $\ddot{\theta}_T$, and third angular rate, $\dddot{\theta}_T$, are virtually zero at the final time for both the nominal and perturbed targets.) For Case 4, the hub angular acceleration shows a small terminal error. However, all other states reach their desired terminal values, including the control and its first derivative. The second derivative of the control torque shows a very small spike near the final time in Figure 3-10. Such spikes are typical when the final gains are large, and may be removed by appropriately adjusting S_f, Q, or α of (3.4.2).

3.7 Extensions

The closed-form solutions given in this paper are only applicable for control problems where the plant dynamics is linear time-invariant. For nonlinear systems, closed-form solutions are much more difficult to obtain. Nevertheless, one may use the following method with the closed-form solutions of this chapter to approximate the solution of the feedback gains for a nonlinear system. First, one obtains the nominal state and control time-histories, using numerical techniques such as shooting methods [67,78] and boundary-value continuation (see Sections 4.3.1 and 4.3.5). Second, the state differential equations are linearized about the nominal solution at discrete points in time. Third, piecewise linear time-invariant intervals are defined about these discrete points in time. Fourth, assuming that the feedback gains are continuous at the boundaries of these intervals, one can transfer the terminal conditions for $\hat{S}(t)$, $\hat{R}(t)$, $\hat{m}(t)$, and $\hat{n}(t)$ into initial conditions by sequentially computing the boundary conditions at each of the interval boundaries, using the

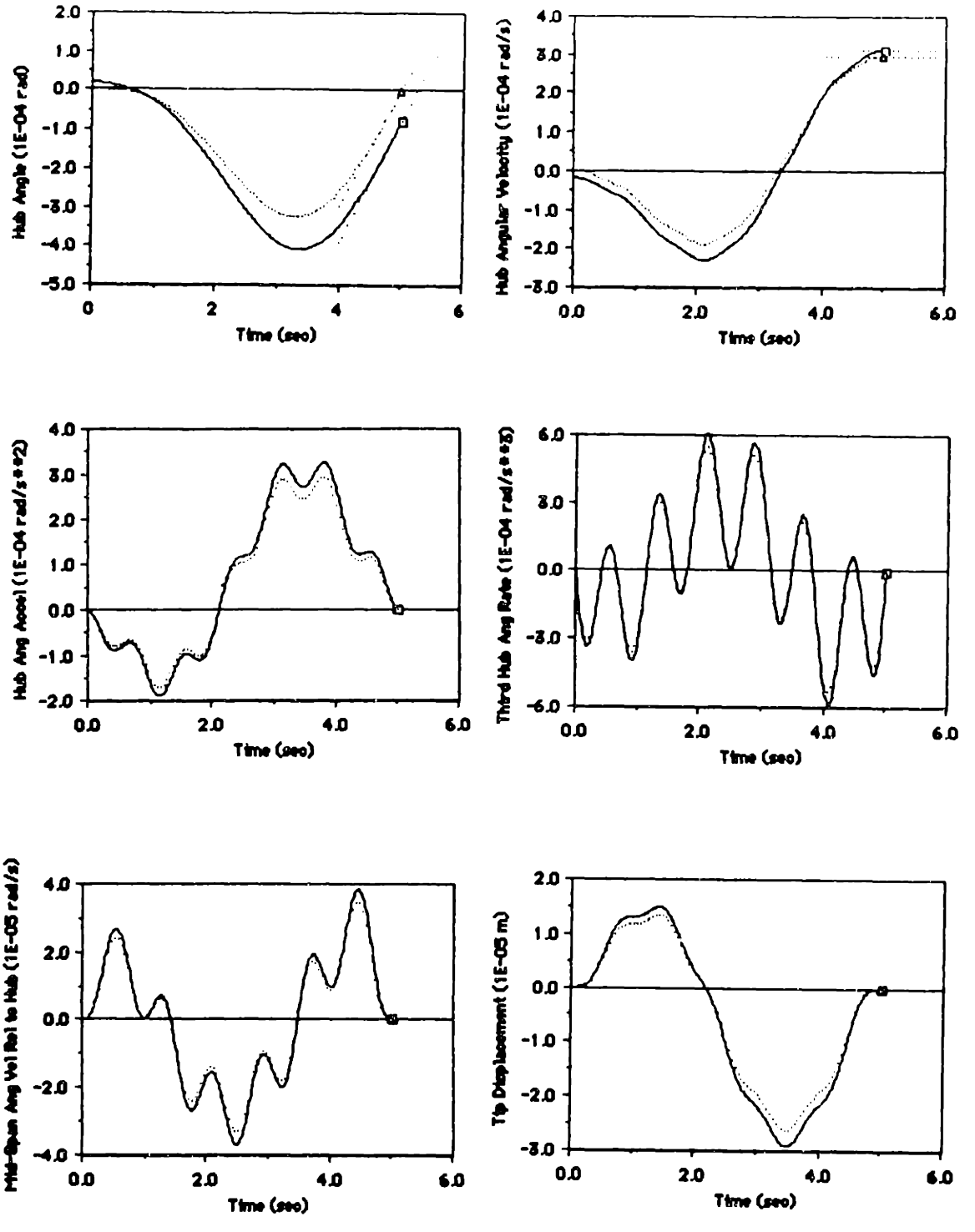


Figure 3-8. Case 3. Retargeting Maneuver

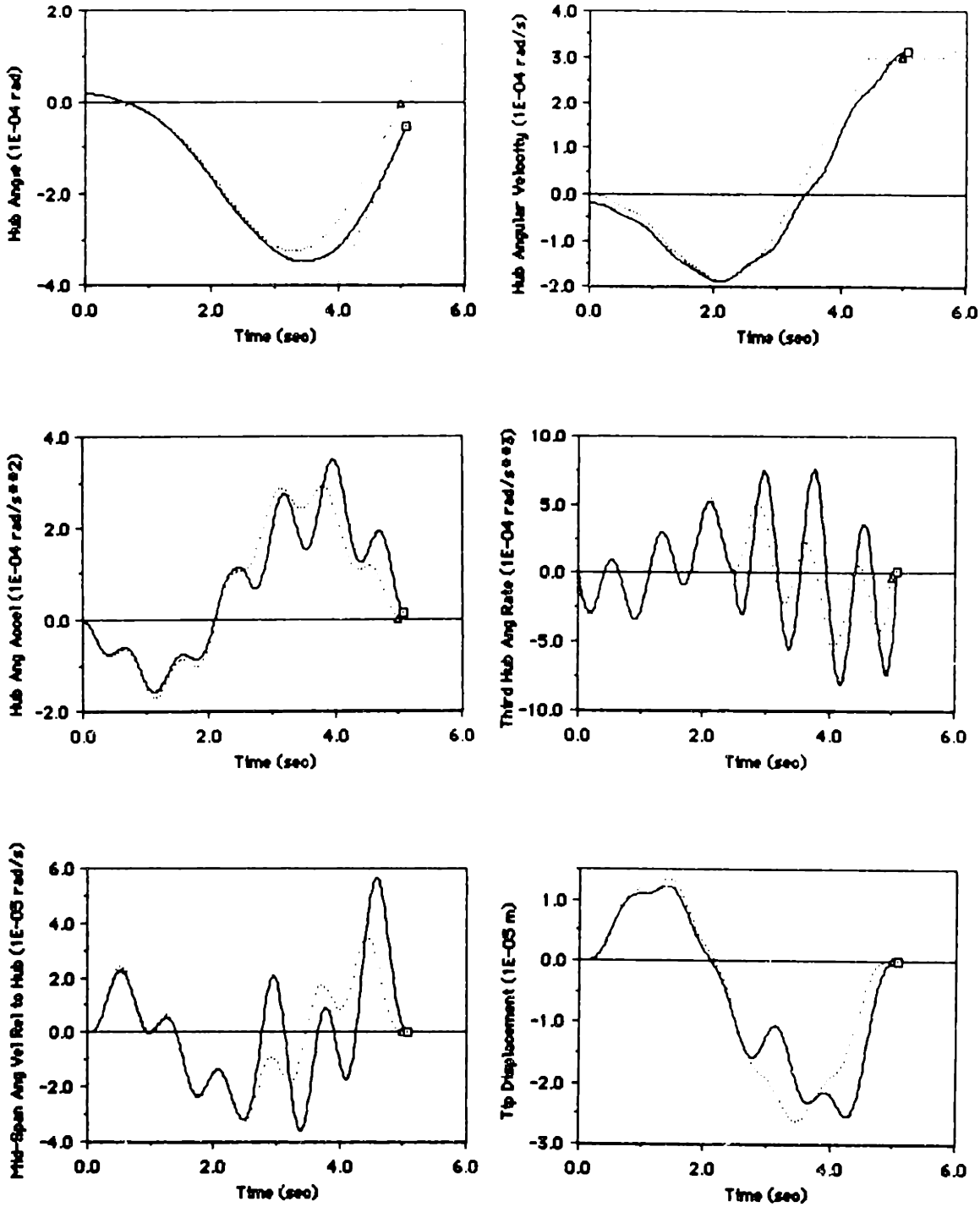
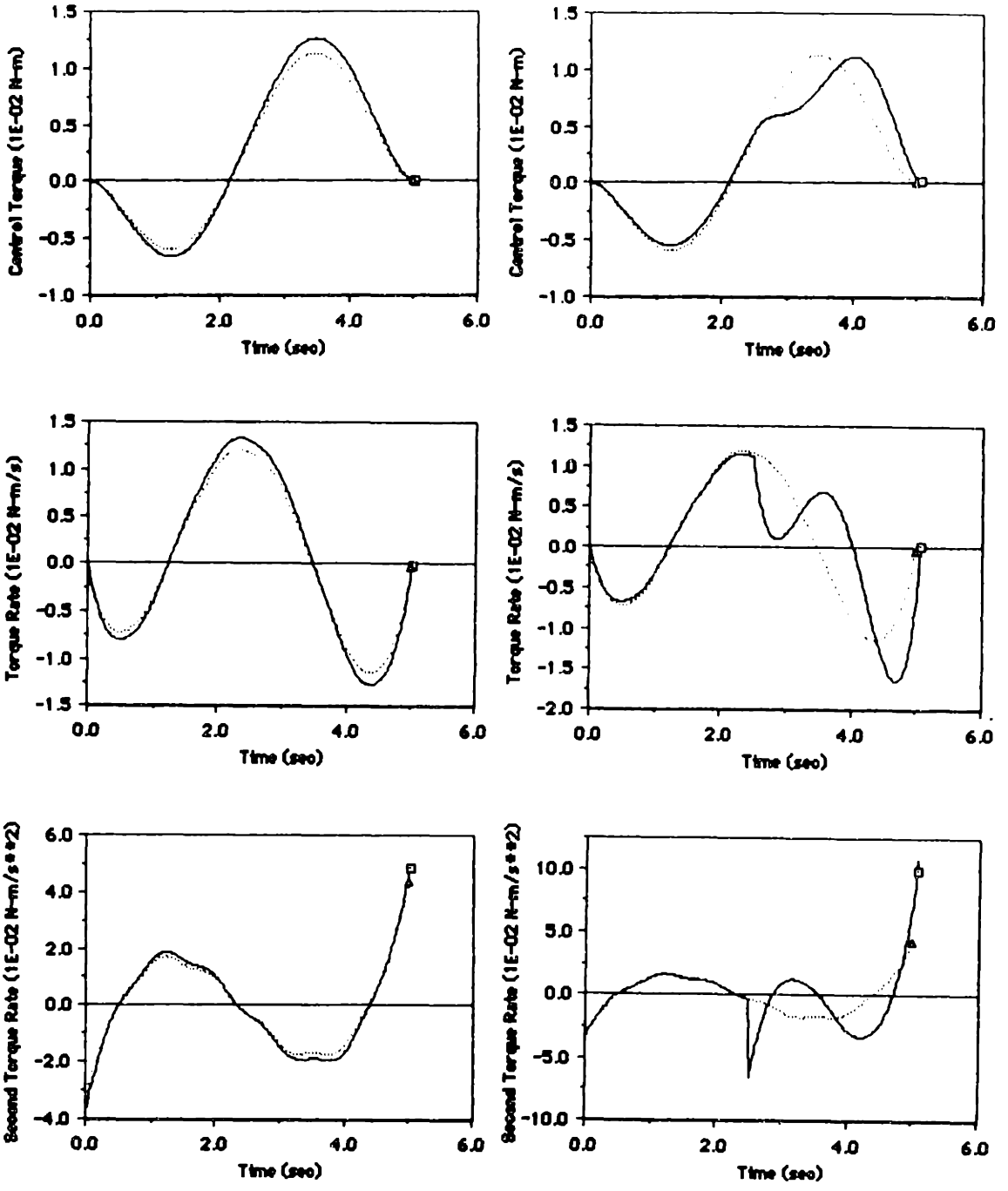


Figure 3-9. Case 4. Retargeting Maneuver with Changing Terminal Constraints



a) Case 3

b) Case 4

Figure 3-10. Control Trajectories for Cases 3 and 4

closed-form solutions, starting from the interval nearest the final time and going backwards in time. Finally, one can use the initial conditions for $\hat{S}(t)$, $\hat{R}(t)$, $\hat{m}(t)$, and $\hat{n}(t)$, with closed-form solutions and piecewise linear time-invariant system differential equations to apply the feedback gains to the perturbed states and perturbed terminal constraints. The length of the linear time-invariant intervals may need to be adjusted depending on the degree of nonlinearity present at a given time along the nominal trajectory.

3.8 Conclusions

Closed-form analytic solutions are presented for the free-final-time perturbation feedback problem with terminal constraints and a linear time-invariant plant. Application of the time-varying feedback gains causes the system to follow a neighboring extremal path when the nominal state trajectory is perturbed by small changes in the initial conditions and terminal constraints. Since the solutions for the feedback gains are obtained in closed form, there is no need for the numerical backward integration of the differential equations for the gains. These solutions are attractive for on-board control implementation because computational errors are not accumulated as much as when integration is used for gain propagation, and the solution is computationally efficient. Numerical experiments indicate that due to the finite precision of digital computers, it is necessary to modify the variables Q and α to obtain good system performance. Example cases demonstrate the utility of the solutions for slewing maneuvers of flexible spacecraft. An extension is proposed for using the closed-form solutions for problems with nonlinear plants.

CHAPTER 4

NONLINEAR THREE-AXIS MANEUVERS FOR FLEXIBLE SPACECRAFT WITH CONTROL SMOOTHING

4.1 Introduction

This chapter presents formulations for general nonlinear three-axis slewing maneuvers for flexible spacecraft. The approach used here is to find the optimal solution for the rigid body model, and then to apply this open-loop rigid body optimal control to the fully flexible spacecraft with a perturbation feedback controller. The perturbation feedback controller controls several flexible modes in addition to the rigid body modes, and the feedback gains are computed using the flexible plant linearized about the rigid body nominal solution at several points along the maneuver. An extended Kalman filter is implemented to estimate the plant states. Example maneuvers are shown using the model of a generic space vehicle.

Section 4.2 presents a discussion of model development and simulation issues. Section 4.3 presents the solution to the nonlinear rigid body problem. The flexible body perturbation formulation is developed in Section 4.4, and the extended Kalman filter is discussed in Section 4.5

4.2 Model Development

The spacecraft model used for the example maneuvers of this chapter is based on a satellite model similar to the N-ROSS satellite, which consists of a more or less rigid bus and several flexible appendages (Figure 4-1). For this study, the spacecraft bus is assumed to be rigid, and only two of the appendages, namely the radiometer and the solar array, are assumed to be flexible.

The frequencies and mode shapes of the flexible appendages are in the form of NASTRAN output data. However, since they are provided by

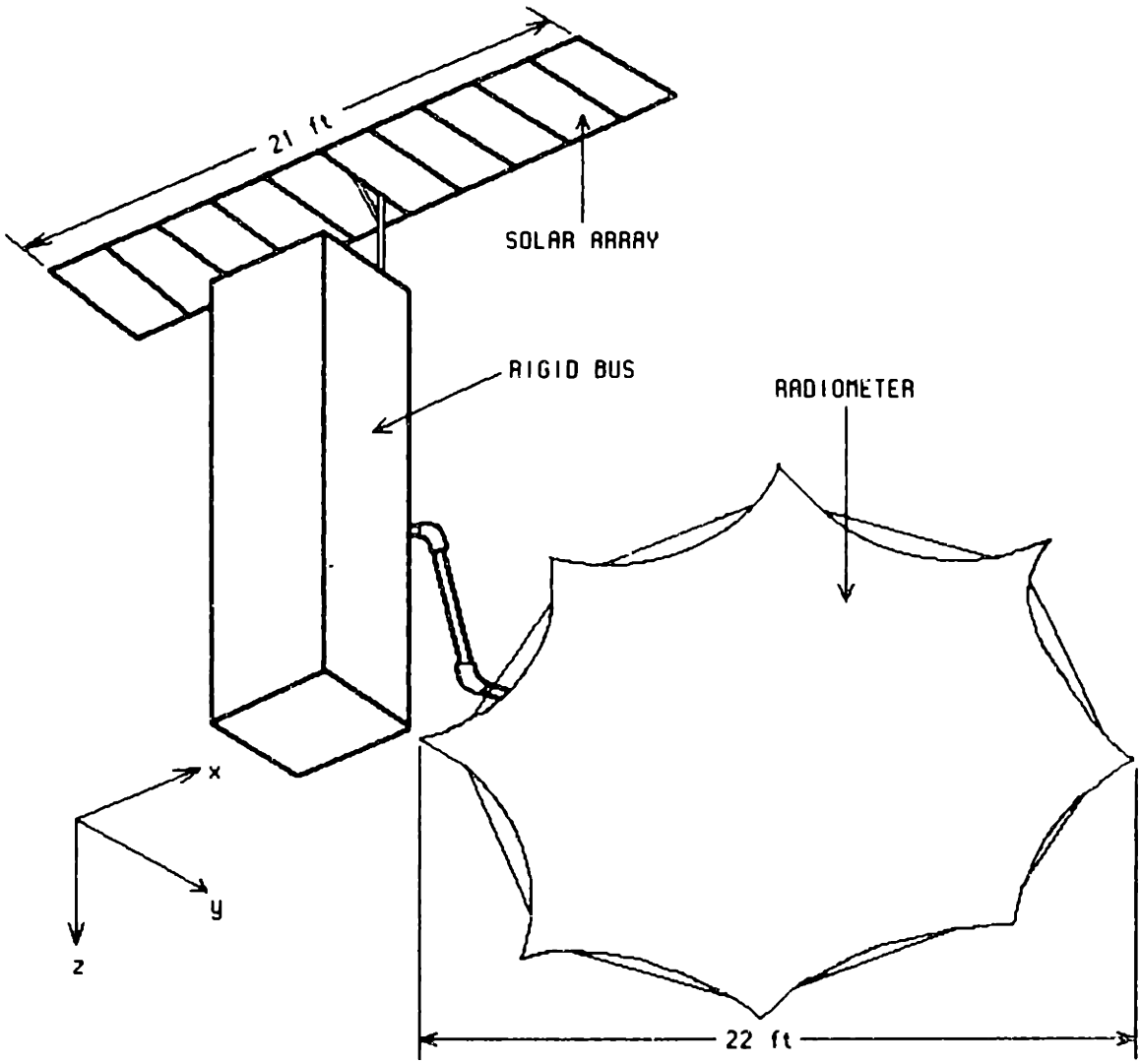


Figure 4-1. Spacecraft Model

different sources, the values need to be scaled to a consistent set of units. Appendix H shows a derivation for the correct scaling of the mode shapes.

The original spacecraft design has one rigid body and six flexible appendages. Since only two of the appendages are considered flexible for this study, the remaining appendages and the rigid body are lumped together to form one rigid body. The total system inertia matrix about the system center of mass is given in units of slug-ft^2 by

$$[I] = \begin{bmatrix} 3888 & -468.7 & 590.7 \\ -468.7 & 4242 & 570.2 \\ 590.7 & 570.2 & 2105 \end{bmatrix} \qquad (4.2.1)$$

The flexible appendages are each assumed to have five elastic degrees of freedom, and their frequencies are listed in Table 4-1. Every mode is assumed to have 0.1% damping.

4.2.1 Multibody Dynamics Simulation

The numerical simulation for the example maneuvers was carried out using a program called DISCOS (Dynamic Interaction Simulation of Controls and Structures) [8], which is a well-known package of software developed for the National Aeronautics and Space Administration (NASA) and distributed by Computer Software Management and Information Center (COSMIC). In DISCOS, a complex structure may be modeled as several rigid or flexible structures connected together at specific points, called hinges. The equations of motion for each body may then be written in the same general form for a single body, with the coupling between bodies provided by Lagrange multipliers which maintain the desired interface constraints.

Table 4-1. Natural Frequencies of Solar Array and Radiometer Modes (Hz)

Mode	Solar Array	Radiometer
1	0.3968	0.5580
2	0.7232	0.6312
3	1.083	0.9080
4	1.368	1.675
5		2.016

The system state vector in DISCOS has the form

$$\{y\} = [\{U\}_1^T \ldots \{U\}_N^T \{\xi\}_1^T \ldots \{\xi\}_N^T \{\beta\}^T \{\delta\}^T]^T , \tag{4.2.2}$$

where

$$\{U\}_i = [\omega_x \ \omega_y \ \omega_z \ u \ v \ w \ \dot{\xi}_1 \ \ldots \ \dot{\xi}_n]_i^T ,$$

ω_x, ω_y, ω_z represent body angular velocities, u, v, w represent the absolute translational velocities in body coordinates, ξ_i represents the modal amplitudes for a flexible body, $\{\beta\}$ represents the vector of unconstrained relative degrees of freedom across hinges, $\{\delta\}$ represents a vector of control variables determined by the user, and N is the number of bodies. The equations of motion are derived from Lagrange's equations, and given by

$$\{\dot{U}\}_j = [m]_j^{-1}(\{G\}_j + [b]_j^T\{\lambda\}), \tag{4.2.3}$$

$$\{\dot{\xi}\}_j = [S_\xi]_j\{U\}_j, \tag{4.2.4}$$

$$\{\dot{\beta}\} = \sum_{j=1}^{N} [B]_j\{U\}_j , \tag{4.2.5}$$

and

$$\{\dot{\delta}\} = f(\{\beta\},\{\dot{\beta}\},\{\xi\},\{\dot{\xi}\},\{\delta\}) , \tag{4.2.6}$$

subject to the constraint equations

$$\sum_{j=1}^{N} [b]_j\{U\}_j = \{\dot{\alpha}\} , \tag{4.2.7}$$

where [m] represents the generalized inertia matrix, {G} represents the generalized force vector, [b] represents kinematic coefficients for the constrained degrees of freedom, $\{\lambda\}$ represents the vector of Lagrange

multipliers, $[S_\xi]$ represents a selection matrix for the modal amplitude rate partition of the $\{U\}$ vector, $[B]$ represents kinematic coefficients for the unconstrained degrees of freedom, and $\{\alpha\}$ represents the constrained relative degrees of freedom at the hinges.

The constraint forces and torques are obtained by differentiating (4.2.7), substituting the result into (4.2.3), and solving for $\{\lambda\}$, yielding

$$\{\lambda\} = \left(\sum_{j=1}^{N} [b]_j [m]_j^{-1} [b]_j^T\right)^{-1} \left[\{\ddot{\alpha}\} - \sum_{j=1}^{N} \left([\dot{b}]_j \{U\}_j + [b]_j [m]_j^{-1} \{G\}_j\right)\right] . \tag{4.2.8}$$

The generalized force vector $\{G\}$ of (4.2.3) is given by the expression

$$\{G\} = \{G_{ex}\} - [-\tfrac{0}{K}-]\{\xi\} - [-\tfrac{0}{C}-]\{\dot{\xi}\} + [\tilde{\Omega}][m]\{U\}$$

$$+ \tfrac{1}{2}\{\{U\}^T [m_{,k}]\{U\}\} - [\dot{m}]\{U\} , \tag{4.2.9}$$

where $\{G_{ex}\}$ is the generalized external force vector, $[K]$ is the modal stiffness matrix, $[C]$ is the modal damping matrix, $[\tilde{\Omega}]$ is a matrix which accounts for gyroscopic and Coriolis effects, and $[m_{,k}]$ represents the partials of the mass matrix with respect to the modal amplitudes.

4.2.2 Recent Issues in Multibody Dynamics Simulation

Doubts have recently been cast on the validity of multibody computer programs such as DISCOS [8], NBOD [31], ALLFLEX [40,41] and TREETOPS [81]. Since DISCOS was chosen to simulate the three-dimensional nonlinear slews of this chapter, an investigation was carried out to determine the validity of such claims [55]. One of the claims was that currently available multibody computer programs do not include rotational inertia terms for the individual elements of a finite element model. This issue may be illustrated as follows. Consider a uniform slender rod of

length l and mass m (Figure 4-2). Its moment of inertia about a lateral axis y through the center of mass is given by $I_{yy} = ml^2/12$. If the rod were divided into two equal segments, the moment of inertia may be approximated by

$$I_{yy} = \int x^2 dm = \sum_{i=1}^{2} m_i x_i^2 , \qquad (4.2.10)$$

where m_i represents the mass of each segment, and x_i represents the center of mass location of each segment with respect to the composite center of mass. For two equal segments, we have $m_i = m/2$ and $x_i = \pm\, l/4$. With these values, the approximate moment of inertia about axis y becomes $I_{yy} = ml^2/16$, which is 30% lower than the correct value. If one associates rotational inertia with the individual segments, then a more accurate expression may be used for the moment of inertia

$$I_{yy} = \sum_{i=1}^{2} m_i x_i^2 + \sum_{i=1}^{2} I_{y_i y_i} , \qquad (4.2.11)$$

where $I_{y_i y_i}$ represent the moment of inertia about an axis parallel to y and intersecting the center of mass of segment i. For two equal segments, we have $I_{y_i y_i} = (1/12)(m/2)(l/2)^2 = ml^2/96$. The value of the expression in (4.2.11) then becomes $I_{yy} = ml^2/12$, which is the correct moment of inertia. Thus, the inclusion of rotational inertia for each individual finite element is important for accurate results, and is crucial for the simulation of torsional elasticity of beams. (The frequency of the torsional mode becomes infinite when rotary inertia is neglected).

4.2.1.1 Finite Element Modelling Issues

The expression for I_{yy} in (4.2.11) is obtained by including rotational kinetic energy, $I_i\omega_i^2/2$, in addition to translational kinetic energy, $m_i v_i^2/2$, in the total kinetic energy expression for the beam of Figure 4-2. For a more general case, consider the finite element shown in Figure 4-3. The rectangle bounded by solid lines represents the undeformed position of a single finite element in a flexible body of a

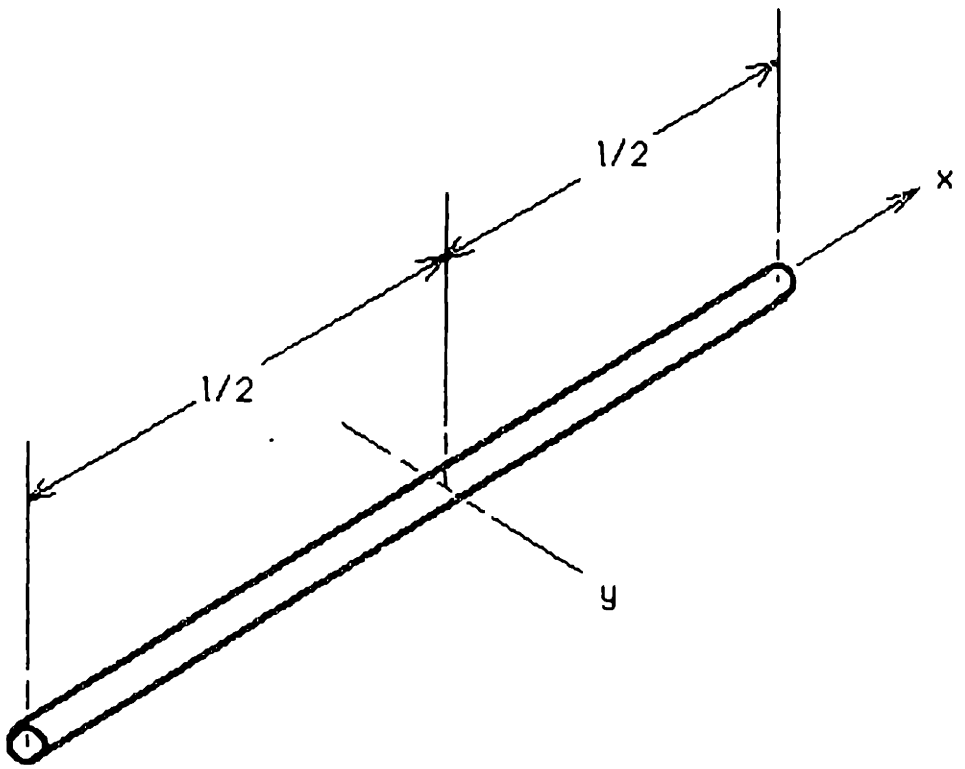


Figure 4-2. Illustration for Moment of Inertia Calculation

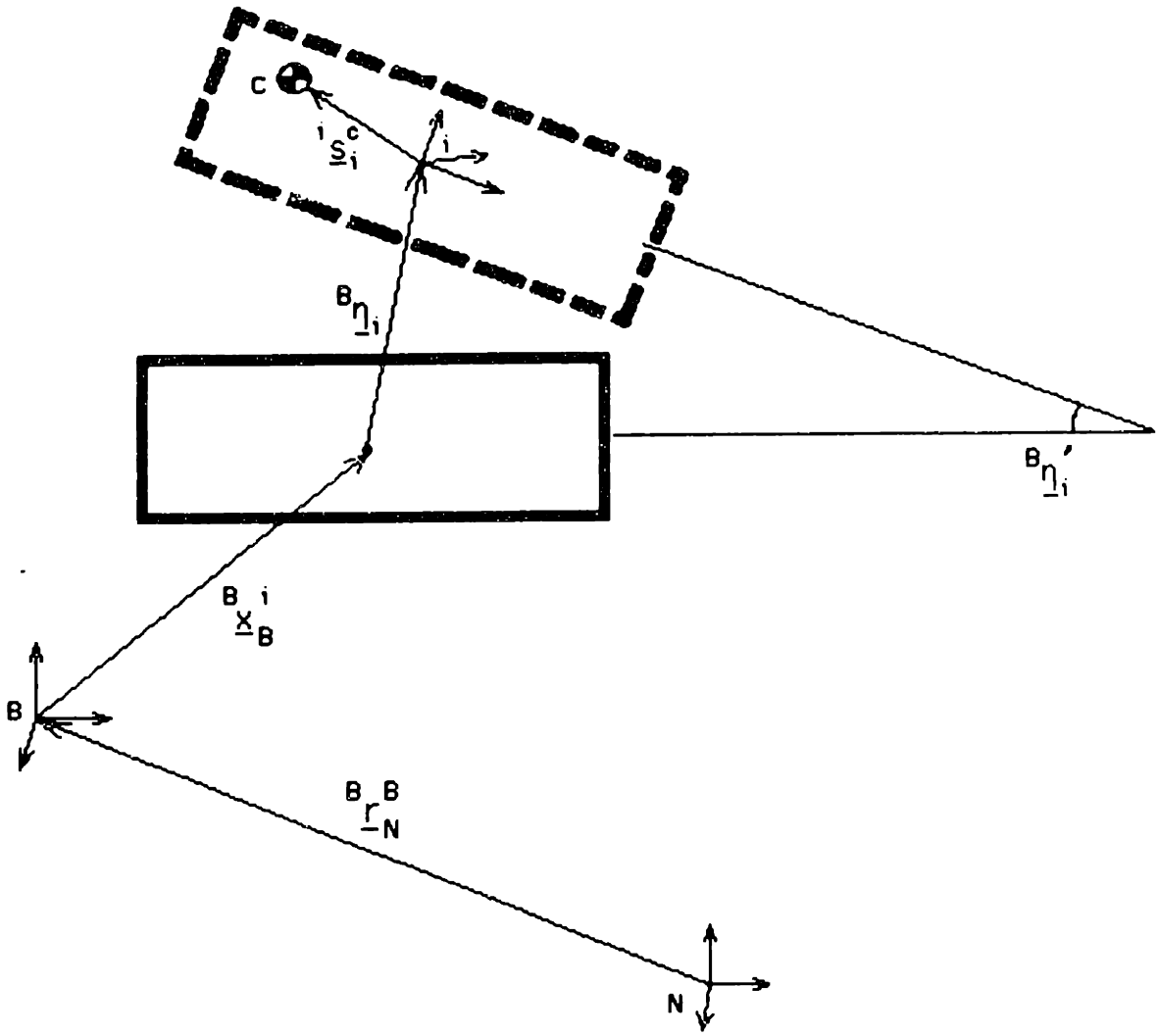


Figure 4-3. Finite-Element Coordinates

multibody system. The rectangle bounded by dashed lines represents the deformed position of the same element. The variables of the type ${}^{a}\underline{r}^{b}_{c}$ represent the vector from point c to point b, resolved in frame a coordinates. In the figure, N represents the inertial frame, B represents the body reference frame, i represents the reference frame of the i-th finite element, and c represents the center of mass of the finite element. The variable ${}^{B}\underline{r}^{B}_{N}$ represents the vector from the inertial frame origin to the body frame origin, and describes the motion of the body in inertial space. The vector ${}^{B}\underline{x}^{i}_{B}$ locates the undeformed position of the i-th finite element within body B, and is constant in B frame coordinates. The variable ${}^{B}\underline{\eta}_{i}$ is the vector from the undeformed position of the i-th finite element frame to its deformed position, and represents the motion of the finite element resulting from the elastic deformation of body B. The variable ${}^{i}\underline{s}^{c}_{i}$ represents the vector from the i-th element frame origin to the center of mass of the element in the deformed position of the element. The vector from the inertial frame origin to the i-th element center of mass, resolved in body frame coordinates, is then given by

$$ {}^{B}\underline{r}^{c}_{N} = {}^{B}\underline{r}^{B}_{N} + {}^{B}\underline{x}^{i}_{B} + {}^{B}\underline{\eta}_{i} + C^{B}_{i}\ {}^{i}\underline{s}^{c}_{i} , \qquad (4.2.12) $$

where C^{B}_{i} is the matrix for transforming vectors in i-th frame coordinates to vectors in B frame coordinates. The elastic deformation, ${}^{B}\underline{\eta}_{i}$, may be written in terms of modal coordinates as

$$ {}^{B}\underline{\eta}_{i} = \sum_{j=1}^{n} \Phi^{i}_{j}\xi_{j} = [h_{i}]\xi , \qquad (4.2.13) $$

where Φ^{i}_{j} represents the mode shape for the j-th mode at the i-th element, ξ_{j} represents the j-th modal amplitude, $[h_{i}]$ and ξ represent the matrix-vector form for the mode shapes and modal amplitudes, and n is the number of modes retained. Similarly, the elastic angular displacement is given by

$$ {}^{B}\underline{\eta}_i' = \sum_{j=1}^{n} (\nabla \times \underline{\phi}_j^i)\xi_j = [\sigma_i]\underline{\xi} , \qquad (4.2.14) $$

where ${}^{B}\underline{\eta}_i'$ represents the angular rotations about the i-th element frame axes, and $\nabla \times \underline{\phi}_j^i$ represents the curl of the mode shape vectors.

In order to obtain the total kinetic energy of the multibody system, one must compute the translational and rotational velocities of all of the elements of each body of the system. The angular velocity of the i-th element may be expressed as the sum of the angular velocities due to overall motion of the body and due to elastic deformation:

$$ {}^{B}\underline{\omega}_N^i = {}^{B}\underline{\omega}_N^B + {}^{B}\underline{\omega}_B^i , \qquad (4.2.15) $$

where ${}^{a}\underline{\omega}_c^b$ represents the angular velocity of frame b with respect to frame c, resolved in frame a coordinates. The second term of (4.2.15) is obtained by differentiating (4.2.14), leading to

$$ {}^{B}\underline{\omega}_N^i = {}^{B}\underline{\omega}_N^B + [\sigma_i]\dot{\underline{\xi}}(t) . \qquad (4.2.16) $$

The translational velocity may be obtained by differentiating (4.2.12) in the inertial frame:

$$ \frac{{}^{B}d}{dt}({}^{B}\underline{r}_N^c)_N = \frac{{}^{B}d}{dt}({}^{B}\underline{r}_N^B)_N + \frac{{}^{B}d}{dt}({}^{B}\underline{x}_B^i)_N + \frac{{}^{B}d}{dt}({}^{B}\underline{\eta}_i)_N + \frac{{}^{B}d}{dt}(C_i^{B}\,{}^{i}\underline{s}_i^c)_N $$

$$ = {}^{B}\underline{v}_N^c , \qquad (4.2.17) $$

where $\frac{{}^{B}d}{dt}(\underline{x})_N$ indicates that the vector $\underline{x}$ is differentiated in the inertial frame, and resolved in body frame coordinates. Using the identity

$$\frac{d}{dt}({}^{a}\underline{x})_b = \frac{d}{dt}({}^{a}\underline{x})_a + \underline{\omega}^{a}_{b} \times {}^{a}\underline{x} \ , \tag{4.2.18}$$

one may re-write (4.2.17) as

$$\begin{aligned} {}^{B}\underline{v}^{c}_{N} = {}^{B}\underline{v}^{B}_{N} + {}^{B}\frac{d}{dt}({}^{B}\underline{x}^{i}_{B})_B + {}^{B}\underline{\omega}^{B}_{N} \times {}^{B}\underline{x}^{i}_{B} + {}^{B}\frac{d}{dt}({}^{B}\underline{\eta}_i)_B + {}^{B}\underline{\omega}^{B}_{N} \times {}^{B}\underline{\eta}_i \\ + {}^{B}\frac{d}{dt}({}^{i}\underline{s}^{c}_{i})_i + {}^{B}\underline{\omega}^{i}_{N} \times {}^{i}\underline{s}^{c}_{i} \ , \end{aligned} \tag{4.2.19}$$

where the last term of (4.2.17) has been resolved in the i-th frame prior to the differentiation. Observing that ${}^{B}\underline{x}^{i}_{B}$ and ${}^{i}\underline{s}^{c}_{i}$ are constant in the B and i-th frames, respectively, and substituting (4.2.13), (4.2.14), and (4.2.16) into (4.2.19), one obtains

$${}^{B}\underline{v}^{c}_{N} = {}^{B}\underline{v}^{B}_{N} - \widetilde{({}^{B}\underline{x}^{i}_{B} + {}^{i}\underline{s}^{c}_{i})}\ {}^{B}\underline{\omega}^{B}_{N} + [h_i]\dot{\underline{\xi}} - \widetilde{{}^{i}\underline{s}^{c}_{i}}[\sigma_i]\dot{\underline{\xi}} - \widetilde{([h_i]\underline{\xi})}\,{}^{B}\underline{\omega}^{B}_{N} \ , \tag{4.2.20}$$

where $\tilde{\underline{x}} = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}$,

and $\underline{x} = [x_1 \ x_2 \ x_3]^T$.

The kinetic energy for all of the elements of body B may then be written in terms of the translational and rotational velocities and inertias:

$$T = \frac{1}{2}\sum_{i=1}^{E}({}^{B}\underline{v}^{c}_{N})^T({}^{B}\underline{v}^{c}_{N})m_i + \frac{1}{2}\sum_{i=1}^{E}({}^{B}\underline{\omega}^{i}_{N})^T[I^{c}_{i}]({}^{B}\underline{\omega}^{i}_{N}) \ , \tag{4.2.21}$$

where E is the number of elements in the body, m_i is the mass of the i-th element, and $[I^{c}_{i}]$ is the rotational inertia matrix for the i-th element

about its center of mass. On substitution of (4.2.16) and (4.2.20) into (4.2.21), the kinetic energy expression becomes

$$
\begin{aligned}
T = \frac{1}{2}\sum_{i=1}^{E} m_i \{ & ({}^B\underline{v}_N^B)^T({}^B\underline{v}_N^B) - ({}^B\underline{v}_N^B)^T({}^B\underline{\tilde{x}}_B^i + {}^i\underline{\tilde{s}}_i^c){}^B\underline{\omega}_N^B \\
& + ({}^B\underline{v}_N^B)^T[h_i]\dot{\underline{\xi}} - ({}^B\underline{v}_N^B)^T\,{}^i\underline{\tilde{s}}_i^c[\sigma_i]\dot{\underline{\xi}} \\
& - ({}^B\underline{v}_N^B)^T(\widetilde{[h_i]\underline{\xi}}){}^B\underline{\omega}_N^B + ({}^B\underline{\omega}_N^B)^T({}^B\underline{\tilde{x}}_B^i + {}^i\underline{\tilde{s}}_i^c){}^B\underline{v}_N^B \\
& - ({}^B\underline{\omega}_N^B)^T({}^B\underline{\tilde{x}}_B^i + {}^i\underline{\tilde{s}}_i^c)({}^B\underline{\tilde{x}}_B^i + {}^i\underline{\tilde{s}}_i^c){}^B\underline{\omega}_N^B \\
& + ({}^B\underline{\omega}_N^B)^T({}^B\underline{\tilde{x}}_B^i + {}^i\underline{\tilde{s}}_i^c)[h_i]\dot{\underline{\xi}} \\
& - ({}^B\underline{\omega}_N^B)^T({}^B\underline{\tilde{x}}_B^i + {}^i\underline{\tilde{s}}_i^c)\,{}^i\underline{\tilde{s}}_i^c[\sigma_i]\dot{\underline{\xi}} \\
& - ({}^B\underline{\omega}_N^B)^T({}^B\underline{\tilde{x}}_B^i + {}^i\underline{\tilde{s}}_i^c)(\widetilde{[h_i]\underline{\xi}}){}^B\underline{\omega}_N^B \\
& + \dot{\underline{\xi}}^T[h_i]^T\,{}^B\underline{v}_N^B - \dot{\underline{\xi}}^T[h_i]^T({}^B\underline{\tilde{x}}_B^i + {}^i\underline{\tilde{s}}_i^c){}^B\underline{\omega}_N^B \\
& + \dot{\underline{\xi}}^T[h_i]^T[h_i]\dot{\underline{\xi}} - \dot{\underline{\xi}}^T[h_i]^T\,{}^i\underline{\tilde{s}}_i^c[\sigma_i]\dot{\underline{\xi}} \\
& - \dot{\underline{\xi}}^T[h_i]^T(\widetilde{[h_i]\underline{\xi}}){}^B\underline{\omega}_N^B + \dot{\underline{\xi}}^T[\sigma_i]^T\,{}^i\underline{\tilde{s}}_i^c\,{}^B\underline{v}_N^B \\
& - \dot{\underline{\xi}}^T[\sigma_i]^T\,{}^i\underline{\tilde{s}}_i^c({}^B\underline{\tilde{x}}_B^i + {}^i\underline{\tilde{s}}_i^c){}^B\underline{\omega}_N^B
\end{aligned}
$$

$$+ \dot{\underline{\xi}}^T[\sigma_i]^T \widetilde{\underline{i}_{s_i}^c}[h_i]\dot{\underline{\xi}} - \dot{\underline{\xi}}^T[\sigma_i]^T \widetilde{\underline{i}_{s_i}^c}\,\widetilde{\underline{i}_{s_i}^c}[\sigma_i]\dot{\underline{\xi}}$$

$$+ \dot{\underline{\xi}}^T[\sigma_i]^T \widetilde{\underline{i}_s^c}(\widetilde{[h_i]\underline{\xi}})\,{}^B\underline{\omega}_N^B + ({}^B\underline{\omega}_N^B)^T(\widetilde{[h_i]\underline{\xi}})\,{}^B\underline{v}_N^B$$

$$- ({}^B\underline{\omega}_N^B)^T(\widetilde{[h_i]\underline{\xi}})(\widetilde{{}^B\underline{x}_B^i + \underline{i}_{s_i}^c})\,{}^B\underline{\omega}_N^B$$

$$+ ({}^B\underline{\omega}_N^B)^T(\widetilde{[h_i]\underline{\xi}})[h_i]\dot{\underline{\xi}} - ({}^B\underline{\omega}_N^B)^T(\widetilde{[h_i]\underline{\xi}})\widetilde{\underline{i}_{s_i}^c}[\sigma_i]\dot{\underline{\xi}}$$

$$- ({}^B\underline{\omega}_N^B)^T(\widetilde{[h_i]\underline{\xi}})(\widetilde{[h_i]\underline{\xi}})\,{}^B\underline{\omega}_N^B \Big\}$$

$$+ \frac{1}{2}\sum_{i=1}^{E}\Big\{({}^B\underline{\omega}_N^B)^T[I_i^c]\,{}^B\underline{\omega}_N^B + ({}^B\underline{\omega}_N^B)^T[I_i^c][\sigma_i]\dot{\underline{\xi}}$$

$$+ \dot{\underline{\xi}}^T[\sigma_i]^T[I_i^c]\,{}^B\underline{\omega}_N^B + \dot{\underline{\xi}}^T[\sigma_i]^T[I_i^c][\sigma_i]\dot{\underline{\xi}}\Big\} . \qquad (4.2.22)$$

Writing the velocity vector for body B as

$$\{U\} = [({}^B\underline{\omega}_N^B)^T\ ({}^B\underline{v}_N^B)^T\ \dot{\underline{\xi}}^T]^T , \qquad (4.2.23)$$

which is identical to the definition of {U} shown following (4.2.2), one can write the kinetic energy for body B in the form

$$T = \frac{1}{2}\{U\}^T[m]\{U\} , \qquad (4.2.24)$$

where the inertia matrix [m] includes translational, rotational, and modal contributions. Specifically, the inertia matrix may be partitioned as follows:

$$
[m] = \left[\begin{array}{ccc|ccc|ccc}
J_{xx} & -J_{xy} & -J_{xz} & 0 & -S_z & S_y & d_{x1} & \cdots & d_{xn} \\
 & J_{yy} & -J_{yz} & S_z & 0 & -S_x & d_{y1} & \cdots & d_{yn} \\
 & & J_{zz} & -S_y & S_x & 0 & d_{z1} & \cdots & d_{zn} \\
\hline
 & & & m & 0 & 0 & a_{x1} & \cdots & a_{xn} \\
 & & & & m & 0 & a_{y1} & \cdots & a_{yn} \\
 & & & & & m & a_{z1} & \cdots & a_{zn} \\
\hline
 & & & & & & e_{11} & \cdots & e_{1n} \\
 & & & & & & & \ddots & \vdots \\
 & & & & & & & & e_{nn}
\end{array}\right] \quad \text{(Symmetric)}
$$

(4.2.25)

where the [J] partition represents the moment of inertia matrix, the [S] partition represents the static moments (due to center of mass offset), the [d] partition represents the coupling between the rotational degrees of freedom and the elastic degrees of freedom, m represents the mass of body B (sum of finite element masses), the [a] partition represents the coupling between the translational and elastic degrees of freedom, and the [e] partition represents the modal inertia matrix (which is often normalized to unity or m).

By examining (4.2.22), one may obtain expressions for the various partitions of [m]. For example, to obtain the J partition of [m], one may look for terms in (4.2.22) which are premultiplied by $({}^B\underline{\omega}_N^B)^T$ and postmultiplied by ${}^B\underline{\omega}_N^B$. As a result, we find

$$
\begin{aligned}
J = \sum_{i=1}^{E} m_i \{ & - (\widetilde{{}^B\underline{x}_B^i + {}^i\underline{s}_i^c})(\widetilde{{}^B\underline{x}_B^i + {}^i\underline{s}_i^c}) - (\widetilde{{}^B\underline{x}_B^i + {}^i\underline{s}_i^c})(\widetilde{[h_i]\underline{\xi}}) \\
& - (\widetilde{[h_i]\underline{\xi}})(\widetilde{{}^B\underline{x}_B^i + {}^i\underline{s}_i^c}) - (\widetilde{[h_i]\underline{\xi}})(\widetilde{[h_i]\underline{\xi}}) \} \\
& + \sum_{i=1}^{E} [I_i^c] \, .
\end{aligned}
\qquad \text{(4.2.26)}
$$

To obtain the element J_{xx}, for example, the calculations indicated by (4.2.26) are performed, and the 1-1 element of the resulting (3 × 3) matrix is extracted, yielding

$$J_{xx} = \sum_{i=1}^{E} \{ I^c_{xxi} + m_i(y_i+s_{yi})^2 + m_i(z_i + s_{zi})^2$$

$$+ 2m_i(z_i + s_{zi})[h_{zi}]\underline{\xi} + 2m_i(y_i + s_{yi})[h_{yi}]\underline{\xi}$$

$$+ m_i\underline{\xi}^T([h_{zi}]^T[h_{zi}] + [h_{yi}]^T[h_{yi}])\underline{\xi} \} , \qquad (4.2.27)$$

where

$$[I^c_i] = \begin{bmatrix} I^c_{xxi} & -I^c_{xyi} & -I^c_{xzi} \\ -I^c_{yxi} & I^c_{yyi} & -I^c_{yzi} \\ -I^c_{zxi} & -I^c_{zyi} & I^c_{zzi} \end{bmatrix} ,$$

$$\underset{\sim}{B}_{\underline{x}_i} = [x_i \; y_i \; z_i]^T ,$$

$$\underline{s}^c_i = [s_{xi} \; s_{yi} \; s_{zi}]^T ,$$

and

$$[h_i] = \begin{bmatrix} [h_{xi}] \\ [h_{yi}] \\ [h_{zi}] \end{bmatrix} . \qquad (3 \times n)$$

Equation (4.2.27) represents the correct expression for J_{xx} with the rotational inertia terms for the individual elements included. Let us now consider what is actually implemented. From Appendix A of Ref [8], and from close examination of the DISCOS source code, one finds that the

inertia matrix [m] of (4.2.25) is computed via a transformation applied to the global inertia matrix:

$$[m] = [\phi]^T[M][\phi] , \tag{4.2.28}$$

where the global inertia matrix (for a lumped mass finite element model) is given by

$$[M] = \begin{bmatrix} [\bar{m}] & [0] & [0] & [0] & [S_z] & -[S_y] \\ [0] & [\bar{m}] & [0] & -[S_z] & [0] & [S_x] \\ [0] & [0] & [\bar{m}] & [S_y] & -[S_x] & [0] \\ [0] & -[S_z] & [S_y] & [I^o_{xx}] & -[I^o_{xy}] & -[I^o_{xz}] \\ [S_z] & [0] & -[S_x] & -[I^o_{yx}] & [I^o_{yy}] & -[I^o_{yz}] \\ -[S_y] & [S_x] & [0] & -[I^o_{zx}] & -[I^o_{zy}] & [I^o_{zz}] \end{bmatrix} , \tag{4.2.29}$$

and the transformation matrix has the form

$$[\phi] = \begin{bmatrix} \{0\} & \{z+[h_z]\xi\} & -\{y+[h_y]\xi\} & \{1\} & \{0\} & \{0\} & [h_x] \\ -\{z+[h_z]\xi\} & \{0\} & \{x+[h_x]\xi\} & \{0\} & \{1\} & \{0\} & [h_y] \\ \{y+[h_y]\xi\} & -\{x+[h_x]\xi\} & \{0\} & \{0\} & \{0\} & \{1\} & [h_z] \\ \{1\} & \{0\} & \{0\} & \{0\} & \{0\} & \{0\} & [\sigma_x] \\ \{0\} & \{1\} & \{0\} & \{0\} & \{0\} & \{0\} & [\sigma_y] \\ \{0\} & \{0\} & \{1\} & \{0\} & \{0\} & \{0\} & [\sigma_z] \end{bmatrix} . \tag{4.2.30}$$

In (4.2.29), each partition within [M] is an (E × E) diagonal matrix, with the i-th diagonal element corresponding to the i-th finite element. The partition $[\bar{m}]$ represents the mass of each finite element; the partitions $[S_x]$, $[S_y]$, and $[S_z]$ represent the static moments for each finite element about their x, y, and z axes which pass through their reference points;

$$I^o_{zzi} = I^c_{zzi} + m_i(s^2_{xi} + s^2_{yi}) ,$$

$$I^o_{xyi} = I^c_{xyi} + m_i s_{xi} s_{yi} ,$$

$$I^o_{xzi} = I^c_{xzi} + m_i s_{xi} s_{zi} ,$$

and

$$I^o_{yzi} = I^c_{yzi} + m_i s_{yi} s_{zi} . \qquad (4.2.31)$$

In addition, the static moments can be expressed as the products of the element masses and their center of mass offsets:

$$S_{xi} = m_i s_{xi} ,$$

$$S_{yi} = m_i s_{yi} ,$$

and

$$S_{zi} = m_i s_{zi} . \qquad (4.2.32)$$

The various partitions of the body inertia matrix [m] may then be obtained by multiplying the appropriate partitions of $[\phi]$ and [M] together, using (4.2.28) through (4.2.32). For comparison with (4.2.27), the element J_{xx}, as computed using (4.2.28) through (4.2.32), is found to be

$$J_{xx} = \{z + [h_z]\underline{\xi}\}^T[\bar{m}]\{z + [h_z]\underline{\xi}\} + \{z + [h_z]\underline{\xi}\}^T[S_z]\{1\}$$

$$+ \{y + [h_y]\underline{\xi}\}^T[\bar{m}]\{y + [h_y]\underline{\xi}\} + \{y + [h_y]\underline{\xi}\}^T[S_y]\{1\}$$

$$+ \{1\}^T[S_z]\{z + [h_z]\xi\} + \{1\}^T[S_y]\{y + [h_y]\xi\}$$

$$+ \{1\}^T[I^o_{xx}]\{1\}$$

$$= \sum_{i=1}^{E} \{ m_i(z_i^2 + 2z_i[h_{zi}]\xi + ([h_{zi}]\xi)^2)$$

$$+ m_i(z_i s_{zi} + s_{zi}[h_{zi}]\xi)$$

$$+ m_i(y_i^2 + 2y_i[h_{yi}]\xi + ([h_{yi}]\xi)^2)$$

$$+ m_i(y_i s_{yi} + s_{yi}[h_{yi}]\xi) + m_i(z_i s_{zi} + s_{zi}[h_{zi}]\xi)$$

$$+ m_i(y_i s_{yi} + s_{yi}[h_{yi}]\xi) + I^c_{xxi} + m_i(s_{yi}^2 + s_{zi}^2) \} . \quad (4.2.33)$$

On comparison between (4.2.27) and the above expression, the two expressions are found to be identical. The same observations have been made for all of the remaining elements in the body inertia matrix [m]. Thus, one can conclude that the inertia matrices used in DISCOS do allow for the inclusion of rotational inertiae of the individual finite elements. The recent concerns about the correctness of the inertia matrix may be traced to the fact that in the text of the DISCOS documentation, the partitions of the inertia matrix are defined as integrals over a continuum. For example,

$$J_{xx} = \int_V [(y + \phi_{yj}\xi_j)^2 + (z + \phi_{zj}\xi_j)^2]\rho \, dV , \quad (4.2.34)$$

where x, y, and z are coordinates of a differential volume element, ϕ_{yj} and ϕ_{zj} are eigenfunctions (mode shape functions) in the y and z directions for the j-th mode, and ρ represents the mass density of a

differential volume element, dV. Without a discussion on the discrete-space implementation of the above expression, one is easily lead to the false conclusion that the discrete-space approximation is given by

$$J_{xx} = \sum_{i=1}^{E} m_i[(y_i + s_{yi} + [h_{yi}]\xi)^2 + (z_i + s_{zi} + [h_{zi}]\xi)^2] \ , \tag{4.2.35}$$

which differs from (4.2.27) and (4.2.33) by the value $\sum_{i=1}^{E} I^c_{xxi}$. Only on careful study of Appendix A of the DISCOS documentation [8] does one realize that the disputed rotational inertia terms are actually accounted for.

4.2.2.2 Arc Length Correction Issues

Another issue of concern is the absence of a gyroscopic stiffening or arc length correction term in current multibody computer programs. The phenomenon and relative importance of gyroscopic stiffening/arc length correction has been discussed in the open literature for many years [42,43,56,64,110]. Specifically, the arc length correction involves the difference between the distance along the longitudinal axis from the end of a long slender rod to a differential mass element in its undeformed position and the distance along the longitudinal axis to the projection of the deformed position of the same mass element. For deformations in one transverse direction, the arc length correction has the form

$$\delta x = \frac{1}{2}\int_0^x [\frac{\partial u(\xi,t)}{\partial \xi}]^2 d\xi \ , \tag{4.2.36}$$

where x is the distance along the longitudinal axis to the undeformed position of the mass element, and u(x,t) is the transverse deformation of the mass element. The inclusion of this arc length correction results in an additional stiffness term in the equations of motion which is

proportional to the square of the angular velocity of the rod, hence the term gyroscopic stiffening.

The correction terms have been implemented in various special purpose codes where the configuration of the structure is fixed. However, general purpose codes do not have provisions for gyroscopic stiffening/arc length correction because not all bodies are made of long slender rods for which the gyroscopic stiffening/arc length correction is important. For a general irregularly shaped body, however, it is impractical, if not impossible, to account for such terms, even if one wanted to.

For the simulation of multibody systems containing long slender rods, one could modify the simulation source codes to include the gyroscopic stiffening/arc length correction terms for the long slender bodies. For general purpose codes, the provision may be made for selectively including gyroscopic stiffening/arc length correction terms to the bodies specified by the user. However, such a provision may entail an additional computational overhead, unless a higher level language is used for producing optimized source code.

4.3 Optimal Nonlinear Three-Dimensional Maneuvers with Control Smoothing for Rigid Structures

4.3.1 Continuation Method

This section presents a solution procedure for obtaining the optimal control profile for rigid body nonlinear three-dimensional slews. The procedure used is the continuation method, also known as the homotopy chain method [76,77,111], whereby a continuation parameter, α, is imbedded into the equations of the problem so that when $\alpha = 0$, the modified equations are easy to solve, and when $\alpha = 1$, the modified equations correspond to the original hard-to-solve problem. By incrementally raising the value of the continuation parameter from 0 to 1, one can numerically carry out an analytic continuation process which starts with the solution of a simple problem and ends with the solution of the difficult problem.

To illustrate the use of the continuation method, let us consider the problem of solving the quadratic equation

$$ax^2 + bx + c = 0 . \tag{4.3.1}$$

Assuming that the solution to

$$ax^2 + c = 0 \tag{4.3.2}$$

is known, one can place the continuation parameter on the linear term of (4.3.1) to produce the family of equations

$$ax^2 + \alpha bx + c = 0 , \tag{4.3.3}$$

so that when $\alpha = 0$, we have (4.3.2), which is easy to solve, and when $\alpha = 1$, we have (4.3.1), which is the original equation to be solved. There are several ways of performing the continuation indicated by (4.3.3). One approach is to obtain the derivative of x with respect to α, and then integrate the derivative from $\alpha = 0$ to $\alpha = 1$, as follows:

$$x(1) = x(0) + \int_0^1 \left(\frac{dx}{d\alpha}\right) d\alpha , \qquad x(0) = \pm \sqrt{-c/a} , \tag{4.3.4}$$

where x is expressed as a function of α. To obtain an expression for $\frac{dx}{d\alpha}$, we differentiate (4.3.3) with respect to α, leading to:

$$2ax \frac{dx}{d\alpha} + bx + \alpha b \frac{dx}{d\alpha} = 0 , \tag{4.3.5}$$

which yields

$$\frac{dx}{d\alpha} = -\frac{bx}{(2ax+\alpha b)} . \tag{4.3.6}$$

To solve for x, we perform the change of variables:

$$x = y - \frac{b\alpha}{2a} , \qquad x(0) = y(0) . \tag{4.3.7}$$

Differentiation of the above equation and substitution into (4.3.6) leads to

$$\frac{dy}{d\alpha} - \frac{b}{2a} = \frac{-by + (b^2\alpha/2a)}{2ay} ,$$

or

$$\frac{dy}{d\alpha} = \frac{b^2\alpha}{4a^2y} . \tag{4.3.8}$$

The above equation is a separable differential equation with solution

$$4a^2\int y\, dy = b^2\int \alpha\, d\alpha ,$$

or

$$2a^2y^2 = \frac{1}{2}b^2\alpha^2 + k . \tag{4.3.9}$$

The value for k is found by setting $\alpha = 0$, using the initial condition for y(0) from (4.3.7), and the solution for x(0) (i.e. $x(0) = \pm\sqrt{-c/a}$) from (4.3.4) to get

$$k = -2ac \,. \qquad (4.3.10)$$

Substitution of (4.3.9) and (4.3.10) into (4.3.7) leads to the expression for x as a function of α:

$$x(\alpha) = -\alpha b/2a \pm \sqrt{\alpha^2 b^2 - 4ac} \,/\, 2a \,. \qquad (4.3.11)$$

When we set $\alpha = 1$, the above equation becomes the well-known solution for a quadratic equation.

The above steps demonstrate the use of analytic continuation, whereby one can obtain a solution to a complex problem. For the above example, we are fortunate to be able to transform the original continuation equation into a separable differential equation which leads to a closed-form solution. In most cases of interest, this is not so. Normally, one must use numerical methods to obtain the solutions. Let us use the above example with $a = 1$, $b = -3$, and $c = -4$. One approach is to numerically integrate (4.3.6), that is, numerically perform the calculation of (4.3.4). Table 4-2a shows the results of numerical integration using a fourth order Runge-Kutta algorithm with several different step-sizes. As shown in the table, a one-step integration is already sufficient for solutions with three digits of accuracy. The table also shows that the error decreases as the number of integration steps is increased, as expected. Table 4-2b shows the intermediate values for a five-step Runge-Kutta integration, corresponding to the last row of Table 4-2a. These intermediate values are actually solutions to (4.3.3) for the intermediate values of α.

Very often, the problem is so complicated that it is difficult to obtain the derivative of the solutions with respect to α. In such a case, one may increase the value of α in small steps, and at each step, iteratively correct the solution so that the modified equation is satisfied for that particular value of α. The increments in α are repeated until $\alpha = 1$ and the original problem is solved. As an example, let us again consider the quadratic equation. Assuming that we have just

Table 4-2a. Summary of Continuation Solutions for Different Number of Runge-Kutta Integration Steps

Number of Steps	Solution 1	Solution 2	Error
1	4.0022956	-1.0022956	±2.30(-3)
2	4.0001014	-1.0001014	±1.01(-4)
3	4.0000184	-1.0000184	±1.84(-5)
4	4.0000056	-1.0000056	±5.61(-6)
5	4.0000022	-1.0000022	±2.25(-6)

Table 4-2b. Intermediate Values for a Five Step Runge-Kutta Integration

Continuation Parameter α	Solution 1	Solution 2
0	2.0000000	-2.0000000
0.2	2.3223751	-1.7223751
0.4	2.6880621	-1.4880621
0.6	3.0931727	-1.2931727
0.8	3.5323827	-1.1323827
1.0	4.0000022	-1.0000022

incremented the value of α, one can define the error in satisfaction of the modified quadratic equation as

$$f(x) = ax^2 + \alpha bx + c \ . \qquad (4.3.12)$$

To reduce the error f(x), one can use a Newton-Raphson iteration [25,32,35], where the correction for x is

$$\Delta x = - f / \left(\frac{df}{dx}\right)$$

$$= - (ax^2 + \alpha bx + c)/(2ax + \alpha b) \ . \qquad (4.3.13)$$

A numerical example is summarized in Table 4-3, where we have used the parameters a = 1, b = -3, and c = -4. The continuation parameter is incremented in five discrete steps, with four to five Newton-Raphson iterations after each increment to achieve eight digits of accuracy. The intermediate values of the solutions may be compared with those in Table 4-2b, which shows slight numerical errors in the Runge-Kutta intermediate values. Of course, the values obtained using numerical integration may also be improved by using Newton-Raphson iterations.

The choice of the step-size in incrementing α is problem-dependent. Too large a step-size may result in non-convergence during the Newton-Raphson iteration; too small a step-size results in higher computational cost. Often, it seems to be advantageous to initially use smaller step-sizes, and then increase the step-size towards the end of the continuation process. (See [26,79]).

For the rigid body nonlinear three-dimensional slewing problem, we use the last approach illustrated above, that is, discrete steps in α combined with Newton-Raphson iterations.

Table 4-3. Summary of Discrete-Step Continuation with Newton-Raphson Iterations

Continuation Parameter α	Solution 1	Solution 2	Number of Iterations†
0	2.0000000	-2.0000000	-
0.2	2.3223748	-1.7223748	4
0.4	2.6880613	-1.4880613	5
0.6	3.0931712	-1.2931712	5
0.8	3.5323808	-1.1323808	5
1.0	4.0000000	-1.0000000	5

†Eight digits of accuracy is the convergence criterion.

4.3.2 Equations of Motion

For the rigid body control problem, let us select as state variables Euler parameters, β, body angular velocities, ω, pseudo-controls, u_o, and pseudo-control rates, u_1. The pseudo-control vector is defined as

$$u_o = [I]^{-1} \begin{Bmatrix} u_x \\ u_y \\ u_z \end{Bmatrix}, \tag{4.3.14}$$

where

$$[I] = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix},$$

and u_x, u_y, and u_z are torques about the x, y, and z axes, respectively. The torques are assumed to be applied by concentrated torque generating devices acting on the rigid spacecraft bus. The inclusion of the pseudo-controls and pseudo-control rates is for control smoothing, as described in Appendix C. Pseudo-controls are used instead of the actual applied torques because the use of applied torques results in large values for the angular velocity costates when the moments of inertia are large. By using pseudo-controls, the problem is normalized so that the values of the states and costates are close to the same order of magnitude. The equations of motion can be shown to be

$$\dot{\beta} = \Omega(\omega)\beta , \qquad (4 \times 1) \tag{4.3.15}$$

$$\dot{\omega} = u_o + G(\omega) , \qquad (3 \times 1) \tag{4.3.16}$$

$$\dot{u}_o = u_1 \,, \qquad (3 \times 1) \qquad (4.3.17)$$

and

$$\dot{u}_1 = u_2 \,, \qquad (3 \times 1) \qquad (4.3.18)$$

where

$$\Omega(\omega) = \frac{1}{2}\begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} ,$$

$$G(\omega) = -[I]^{-1}[\tilde{\omega}][I]\omega \,,$$

$$\omega = [\omega_x \ \omega_y \ \omega_z]^T \,,$$

and

$$[\tilde{\omega}] = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} .$$

In the above equations, (4.3.15) is the kinematic equation relating the Euler parameter rates and the body angular velocities, and (4.3.16) represents Euler's equation in terms of pseudo-controls.

4.3.3 Optimal Control Problem and Necessary Conditions

For rigid body nonlinear three-dimensional slews, let us define the

optimal control problem as the minimization of a finite-time quadratic performance index

$$J = \frac{1}{2}\int_{t_o}^{t_f}[\chi^T(t)\ u_2^T(t)]\ W\left\{\begin{array}{c}\chi(t)\\ u_2(t)\end{array}\right\}dt\ , \qquad (4.3.19)$$

where

$$\chi = [\beta^T\ \omega^T\ u_o^T\ u_1^T]^T\ ,$$

and

$$W = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & Q & 0 & 0 & 0 \\ 0 & 0 & I & 0 & I/\omega_B^2 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I/\omega_B^2 & 0 & I/\omega_B^4 \end{bmatrix}\ ,$$

subject to the state dynamics equations, (4.3.15) through (4.3.18), with specified initial and final states. The symbol I represents the (3 × 3) identity matrix. In the above performance index of (4.3.19), the weight matrix, W, does not include penalties on the Euler parameters, since the angular displacements may be large. A weighting matrix is placed on the angular velocity terms so that the angular velocities may be kept small. The penalty terms on the pseudo-controls and pseudo-control rates are the time-domain equivalent of frequency-domain penalties on the pseudo-control, where the frequency range above ω_B is penalized. (See Appendix C).

The Hamiltonian for the performance index of (4.3.19) and the state dynamics of (4.3.15) through (4.3.18) may be written as

$$H = \frac{1}{2}[\omega^T Q\omega + u_o^T u_o + u_2^T u_2/\omega_B^4 + 2u_o^T u_2/\omega_B^2]$$

$$+ \gamma^T \Omega(\omega)\beta + \lambda^T(u_o + G(\omega)) + \mu_o^T u_1 + \mu_1^T u_2 , \qquad (4.3.20)$$

where γ (4 × 1), λ (3 × 1), μ_o (3 × 1), and μ_1 (3 × 1) are costate or adjoint variables for β, ω, u_o, and u_1, respectively. The necessary conditions can then be derived from the Hamiltonian as

$$[\frac{\partial H}{\partial \gamma}]^T = \dot{\beta} = \Omega(\omega)\beta , \qquad (4.3.21)$$

$$[\frac{\partial H}{\partial \lambda}]^T = \dot{\omega} = u_o + G(\omega) , \qquad (4.3.22)$$

$$[\frac{\partial H}{\partial \mu_o}]^T = \dot{u}_o = u_1 , \qquad (4.3.23)$$

$$[\frac{\partial H}{\partial \mu_1}]^T = \dot{u}_1 = u_2 , \qquad (4.3.24)$$

$$- [\frac{\partial H}{\partial \beta}]^T = \dot{\gamma} = \Omega(\omega)\gamma , \qquad (4.3.25)$$

$$- [\frac{\partial H}{\partial \omega}]^T = \dot{\lambda} = - Q\omega - [\frac{\partial}{\partial \omega}(\gamma^T \Omega\beta)]^T - [\frac{\partial G}{\partial \omega}]^T\lambda , \qquad (4.3.26)$$

$$- [\frac{\partial H}{\partial u_o}]^T = \dot{\mu}_o = - u_o - u_2/\omega_B^2 - \lambda , \qquad (4.3.27)$$

$$-\left[\frac{\partial H}{\partial u_1}\right]^T = \dot{\mu}_1 = -\mu_o \,, \tag{4.3.28}$$

and

$$\left[\frac{\partial H}{\partial u_2}\right]^T = 0 = u_2/\omega_B^4 + u_o/\omega_B^2 + \mu_1 \,, \tag{4.3.29}$$

where the following notation has been used

$$\left[\frac{\partial v}{\partial w}\right]_{ij} = \frac{\partial v_i}{\partial w_j} \,,$$

for general vectors (or scalars) v and w. In the equations above, the initial and final values for β, ω, u_o and u_1 are specified. However, no boundary conditions are known for γ, λ, μ_o, and μ_1. Performing the partial derivatives indicated in (4.3.26), one can write the differential equation for λ as

$$\dot{\lambda} = -Q\omega - B^T\gamma + \{\widetilde{[I\omega]} - [I][\tilde{\omega}]\}[I]^{-1}\lambda \,,$$

or

$$\dot{\lambda} = -Q\omega + \Gamma^T\beta + \{\widetilde{[I\omega]} - [I][\tilde{\omega}]\}[I]^{-1}\lambda \,, \tag{4.3.30}$$

where

$$B = \frac{1}{2}\begin{bmatrix} -\beta_1 & -\beta_2 & -\beta_3 \\ \beta_o & -\beta_3 & \beta_2 \\ \beta_3 & \beta_o & -\beta_1 \\ -\beta_2 & \beta_1 & \beta_o \end{bmatrix} \,,$$

$$\Gamma = \frac{1}{2}\begin{bmatrix} -\gamma_1 & -\gamma_2 & -\gamma_3 \\ \gamma_o & -\gamma_3 & \gamma_2 \\ \gamma_3 & \gamma_o & -\gamma_1 \\ -\gamma_2 & \gamma_1 & \gamma_o \end{bmatrix} ,$$

$$\beta = [\beta_o \ \beta_1 \ \beta_2 \ \beta_3]^T ,$$

and

$$\gamma = [\gamma_o \ \gamma_1 \ \gamma_2 \ \gamma_3]^T .$$

The alternative equation forms in (4.3.30) are particularly useful for computing the state transition matrix partial derivatives. From (4.3.29), the commanded second time derivative of the pseudo-control may be written as

$$u_2 = -\omega_B^4[\mu_1 + u_o/\omega_B^2] . \qquad (4.3.31)$$

4.3.4 Starting Guess for the Continuation Method

The necessary conditions for the rigid body slewing problem shown in (4.3.21) through (4.3.29) represent a set of difficult nonlinear differential equations with split boundary conditions. To solve the differential equations, a continuation method is used, as discussed in Section 4.3.1. The first step is to find a starting guess which is easy to solve. For this, we choose a single-axis maneuver about a principle axis. To further simplify the calculations, we assume initially that the inertia matrix of the spacecraft is diagonal, with the non-zero off-diagonal terms introduced during the continuation process.

A reasonable choice of axis for the starting guess solution is to use the axis with the highest peak angular momentum if the single-axis maneuver were to be accomplished via bang-bang control. This is also the

axis about which the largest bang-bang torque would be applied. Denoting this axis by k, where k = 1, 2, or 3 corresponds to the x, y, or z axis, one may write a subset of the necessary conditions of (4.3.21) through (4.3.29) for rotation about axis k:

$$\dot{\beta}_o = -\frac{1}{2}\omega_k\beta_k \,, \tag{4.3.32}$$

$$\dot{\beta}_k = \frac{1}{2}\omega_k\beta_o \,, \tag{4.3.33}$$

$$\dot{\omega}_k = (u_o)_k \,, \tag{4.3.34}$$

$$(\dot{u}_o)_k = (u_1)_k \,, \tag{4.3.35}$$

$$(\dot{u}_1)_k = -\omega_B^2(u_o)_k - \omega_B^4(\mu_1)_k \,, \tag{4.3.36}$$

$$\dot{\gamma}_o = -\frac{1}{2}\omega_k\gamma_k \,, \tag{4.3.37}$$

$$\dot{\gamma}_k = \frac{1}{2}\omega_k\gamma_o \,, \tag{4.3.38}$$

$$\dot{\lambda}_k = -Q_k\omega_k - \frac{1}{2}\gamma_k\beta_o + \frac{1}{2}\gamma_o\beta_k \,, \tag{4.3.39}$$

$$(\dot{\mu}_o)_k = -\lambda_k + \omega_B^2(\mu_1)_k \,, \tag{4.3.40}$$

and

$$(\dot{\mu}_1)_k = -(\mu_o)_k \,, \tag{4.3.41}$$

where (4.3.31) has been substituted into (4.3.24) and (4.3.27), and it is assumed that the angular orientations about the other axes, $j \neq k$, are equal to zero for the Euler parameter equation.

The solution for the Euler parameters is obtained as follows. Let us write (4.3.32) and (4.3.33) in matrix-vector form:

$$\begin{Bmatrix} \dot{\beta}_o(t) \\ \dot{\beta}_k(t) \end{Bmatrix} = \begin{bmatrix} 0 & -\frac{1}{2}\omega_k(t) \\ \frac{1}{2}\omega_k(t) & 0 \end{bmatrix} \begin{Bmatrix} \beta_o(t) \\ \beta_k(t) \end{Bmatrix} . \qquad (4.3.42)$$

Since the matrix coefficient of the above equation is time-varying, the solution for (4.3.42) is

$$\begin{Bmatrix} \beta_o(t) \\ \beta_k(t) \end{Bmatrix} = \exp \begin{bmatrix} 0 & -\frac{1}{2}(\phi_k(t)-\phi_k(t_o)) \\ \frac{1}{2}(\phi_k(t)-\phi_k(t_o)) & 0 \end{bmatrix} \begin{Bmatrix} \beta_o(t_o) \\ \beta_k(t_o) \end{Bmatrix} , \qquad (4.3.43)$$

where

$$\phi_k(t)-\phi_k(t_o) = \int_{t_o}^{t} \omega_k(\tau)d\tau$$

is the angular displacement about axis k. On exponentiation of the matrix in (4.3.43), the solution for the Euler parameters is

$$\beta_o(t) = \cos(\theta_k/2)\beta_o(t_o) - \sin(\theta_k/2)\beta_k(t_o) , \qquad (4.3.44)$$

and

$$\beta_k(t) = \sin(\theta_k/2)\beta_o(t_o) + \cos(\theta_k/2)\beta_k(t_o) , \qquad (4.3.45)$$

where

$$\theta_k(t) = \phi_k(t) - \phi_k(t_o) .$$

From tables relating Euler angles to Euler parameters, we find for the single axis case

$$\beta_o(t_o) = \cos(\phi_k(t_o)/2) , \tag{4.3.46}$$

and

$$\beta_k(t_o) = \sin(\phi_k(t_o)/2) . \tag{4.3.47}$$

Substitution of (4.3.46) and (4.3.47) into (4.3.44) and (4.3.45) leads to

$$\beta_o(t) = \cos(\phi_k(t)/2) , \tag{4.3.48}$$

and

$$\beta_k(t) = \sin(\phi_k(t)/2) . \tag{4.3.49}$$

Note that the above solution could easily have been obtained from the definition of the Euler parameters. However, (4.3.44) and (4.3.45) are useful for solving the Euler parameter costates, γ_o and γ_k, which have the same differential equation as the Euler parameters. The result of replacing β by γ in (4.3.44) and (4.3.45) is

$$\gamma_o(t) = \cos(\theta_k/2)\gamma_o(t_o) - \sin(\theta_k/2)\gamma_k(t_o) , \tag{4.3.50}$$

and

$$\gamma_k(t) = \sin(\theta_k/2)\gamma_o(t_o) + \cos(\theta_k/2)\gamma_k(t_o) . \tag{4.3.51}$$

Since there are three degrees of freedom for a general angular motion while there are four Euler parameters, the Euler parameters are made unique by imposing the sum-squared constraint:

$$\sum_{i=1}^{4} \beta_i^2 = 1 . \tag{4.3.52}$$

Similarly, a constraint must be imposed on the Euler parameter costates to provide uniqueness. We use an orthogonality constraint [106]:

$$\sum_{i=1}^{4} \beta_i \gamma_i = 0 \ . \qquad (4.3.53)$$

From (4.3.46), (4.3.47), and (4.3.53), one finds

$$\gamma_o(t_o) = - \tan(\phi_k(t_o)/2)\gamma_k(t_o) \ . \qquad (4.3.54)$$

From (4.3.50), (4.3.51), and (4.3.54), the solution for the Euler parameter costates is

$$\gamma_o(t) = - C\sin(\phi_k(t)/2) \ , \qquad (4.3.55)$$

and

$$\gamma_k(t) = C\cos(\phi_k(t)/2) \ , \qquad (4.3.56)$$

where

$$C = \gamma_k(t_o)\sec(\phi_k(t_o)/2) \ .$$

The solution for $\lambda(t)$ may be obtained by substituting (4.3.48), (4.3.49), (4.3.55), and (4.3.56) into (4.3.39) to give

$$\dot{\lambda}_k(t) = - Q_k\omega_k(t) - \frac{1}{2} C \ , \qquad (4.3.57)$$

which yields

$$\lambda_k(t) = - Q_k(\phi_k(t)-\phi_k(t_o)) - \frac{1}{2} C(t-t_o) + \lambda_k(t_o) \ . \qquad (4.3.58)$$

Using (4.3.34) through (4.3.36), (4.3.40), (4.3.41), (4.3.57), (4.3.58), and the definition for $\phi_k(t)$ following (4.3.43), and defining

$$\xi(t) = \lambda_k(t_o) - \frac{1}{2} C(t-t_o) , \tag{4.3.59}$$

one can write the single-axis system state equations in the following form:

$$\frac{d}{dt} \begin{Bmatrix} \phi_k(t)-\phi_k(t_o) \\ \omega_k(t) \\ (u_o)_k(t) \\ (u_1)_k(t) \\ (\mu_o)_k(t) \\ (\mu_1)_k(t) \\ \xi(t) \\ C \end{Bmatrix} = [K] \begin{Bmatrix} \phi_k(t)-\phi_k(t_o) \\ \omega_k(t) \\ (u_o)_k(t) \\ (u_1)_k(t) \\ (\mu_o)_k(t) \\ (\mu_1)_k(t) \\ \xi(t) \\ C \end{Bmatrix} , \tag{4.3.60}$$

where

$$[K] = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\omega_B^2 & 0 & 0 & -\omega_B^4 & 0 & 0 \\ Q_k & 0 & 0 & 0 & 0 & \omega_B^2 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} .$$

To obtain the unknown initial costates, the final states are written in terms of the initial states via the state transition matrix:

$$\begin{Bmatrix} \phi_k(t_f)-\phi_k(t_o) \\ \omega_k(t_f) \\ (u_o)_k(t_f) \\ (u_1)_k(t_f) \\ (\mu_o)_k(t_f) \\ (\mu_1)_k(t_f) \\ \xi(t_f) \\ C \end{Bmatrix} = e^{K(t_f-t_o)} \begin{Bmatrix} 0 \\ \omega_k(t_o) \\ (u_o)_k(t_o) \\ (u_1)_k(t_o) \\ (\mu_o)_k(t_o) \\ (\mu_1)_k(t_o) \\ \xi(t_o) \\ C \end{Bmatrix} . \tag{4.3.61}$$

Since the initial and final values for ϕ_k, ω_k, $(u_o)_k$, and $(u_1)_k$ are known, the first four rows of (4.3.61) may be re-arranged in partitioned matrix-vector form to yield the solution for the unknown initial costates:

$$\Theta_{12} \begin{Bmatrix} (\mu_o)_k(t_o) \\ (\mu_1)_k(t_o) \\ \xi(t_o) \\ C \end{Bmatrix} = \begin{Bmatrix} \phi_k(t_f)-\phi_k(t_o) \\ \omega_k(t_f) \\ (u_o)_k(t_f) \\ (u_1)_k(t_f) \end{Bmatrix} - \Theta_{11} \begin{Bmatrix} 0 \\ \omega_k(t_o) \\ (u_o)_k(t_o) \\ (u_1)_k(t_f) \end{Bmatrix} , \tag{4.3.62}$$

where

$$e^{K(t_f-t_o)} = \begin{bmatrix} \Theta_{11} & \Theta_{12} \\ \Theta_{21} & \Theta_{22} \end{bmatrix} .$$

Equation (4.3.62) may be solved for $(\mu_o)_k(t_o)$, $(\mu_1)_k(t_o)$, $\xi(t_o)$, and C via Gaussian elimination. The initial angular velocity costate is then identified from (4.3.59) as

$$\lambda_k(t_o) = \xi(t_o) ; \tag{4.3.63}$$

and the initial Euler parameter costates are computed from (4.3.55) and (4.3.56), giving

$$\gamma_o(t_o) = -C\sin\left(\phi_k(t_o)/2\right) , \tag{4.3.64}$$

and

$$\gamma_k(t_o) = C\cos\left(\phi_k(t_o)/2\right) . \tag{4.3.65}$$

Using the initial states $\phi_k(t_o)$, $\omega_k(t_o)$, $(u_o)_k(t_o)$, and $(u_1)_k(t_o)$, the initial costates from (4.3.62) through (4.3.65), and the state-costate differential equations of (4.3.32) through (4.3.41), one can generate the optimal trajectory minimizing (4.3.19) for the single-axis diagonal inertia matrix special case.

4.3.5 Continuation for Inertia Matrix and Boundary Conditions

Given the single-axis rotation with diagonal inertia matrix starting guess of Section 4.3.4, the three-axis optimal maneuver with fully populated inertia matrix may be obtained through a continuation process. The continuation approach used is the one described in the third example of Section 4.3.1, where the continuation parameter, α, is incremented at discrete steps, with convergence at each step achieved via a Newton-Raphson iteration.

Since there are two different quantities introduced during the continuation process -- off-diagonal elements of the inertia matrix and three-axis boundary conditions -- the continuation process may be performed separately, or combined together in one process. When one combined continuation process is used, it may be advantageous to retain the ability to use separate continuation parameter increments for the two quantities when handling extremely difficult problems.

For the inertia matrix continuation, the inertia matrix of (4.3.14) is replaced by

$$[I(\alpha_1)] = \begin{bmatrix} I_{xx} & -\alpha_1 I_{xy} & -\alpha_1 I_{xz} \\ -\alpha_1 I_{yx} & I_{yy} & -\alpha_1 I_{yz} \\ -\alpha_1 I_{zx} & -\alpha_1 I_{zy} & I_{zz} \end{bmatrix} . \qquad (4.3.66)$$

Setting $\alpha_1 = 0$ produces the diagonal inertia matrix used in Section 4.3.4, and setting $\alpha_1 = 1$ produces the original fully populated inertia matrix.

For the boundary condition continuation, let us define the modified terminal Euler angles as

$$\psi_j(t_f,\alpha_2) = \begin{cases} \phi_j(t_f) , & j = k \\ \alpha_2\phi_j(t_f) , & j \neq k \end{cases} \qquad (4.3.67)$$

where k represents the axis used for the starting guess of Section 4.3.4, and $\phi_j(t_f)$, (j=1,2,3), are the desired final Euler angles of the three-axis maneuver. For each value of the continuation parameter, α_2, the final Euler parameters are computed from the values of ψ_j. For example, the final Euler parameters for a 1-2-3 Euler angle sequence are given as

$$\beta_0(t_f,\alpha_2) = c(\tfrac{\psi_1}{2})c(\tfrac{\psi_2}{2})c(\tfrac{\psi_3}{2}) - s(\tfrac{\psi_1}{2})s(\tfrac{\psi_2}{2})s(\tfrac{\psi_3}{2}) ,$$

$$\beta_1(t_f,\alpha_2) = s(\tfrac{\psi_1}{2})c(\tfrac{\psi_2}{2})c(\tfrac{\psi_3}{2}) - c(\tfrac{\psi_1}{2})s(\tfrac{\psi_2}{2})s(\tfrac{\psi_3}{2}) ,$$

$$\beta_2(t_f,\alpha_2) = c(\tfrac{\psi_1}{2})s(\tfrac{\psi_2}{2})c(\tfrac{\psi_3}{2}) - s(\tfrac{\psi_1}{2})c(\tfrac{\psi_2}{2})s(\tfrac{\psi_3}{2}) ,$$

and

$$\beta_3(t_f,\alpha_2) = c(\tfrac{\psi_1}{2})c(\tfrac{\psi_2}{2})s(\tfrac{\psi_3}{2}) - s(\tfrac{\psi_1}{2})s(\tfrac{\psi_2}{2})c(\tfrac{\psi_3}{2}) , \qquad (4.3.68)$$

where c and s denote the cosine and sine functions, respectively. The modified final conditions for the angular velocities, pseudo-controls, and

pseudo-control rates are similarly defined as

$$\omega_j(t_f,\alpha_2) = \begin{cases} \omega_j(t_f) & j = k\ , \\ \alpha_2\omega_j(t_f) & j \neq k\ , \end{cases} \tag{4.3.69}$$

$$(u_o)_j(t_f,\alpha_2) = \begin{cases} (u_o)_j(t_f) & j = k\ , \\ \alpha_2(u_o)_j(t_f) & j \neq k\ , \end{cases} \tag{4.3.70}$$

and

$$(u_1)_j(t_f,\alpha_2) = \begin{cases} (u_1)_j(t_f) & j = k\ , \\ \alpha_2(u_1)_j(t_f) & j \neq k\ . \end{cases} \tag{4.3.71}$$

The modified initial conditions are defined in the same manner as for the modified final conditions.

After each increase of the continuation parameters, the previously converged values of the initial costates no longer generate trajectories which satisfy the final boundary conditions. As a result, an iterative correction scheme is needed to correct the initial costates based on the error in satisfying the final boundary conditions. For this purpose, a Newton-Raphson first order correction scheme is used. This is accomplished by Taylor expanding the terminal values of the states as functions of the initial costates. To obtain quicker convergence, one may use extrapolated values of the initial costates based on previously converged values and back α values [18,76,89].

For each iteration, the modified state-costate vector is integrated from t_o to t_f, and the error in satisfaction of the modified final conditions is computed. During the integration of the states and costates, their partial derivatives with respect to the initial costates are also integrated. That is, the matrices

$$\frac{\partial \chi(t)}{\partial \Lambda(t_o)} \qquad \text{and} \qquad \frac{\partial \Lambda(t)}{\partial \Lambda(t_o)}$$

are integrated from t_o to t_f, where

$$\Lambda = [\gamma^T \lambda^T \mu_o^T \mu_f^T]^T , \tag{4.3.72}$$

and χ is defined following (4.3.19). The initial conditions and differential equations for integrating these partials are presented in Appendix I. The corrections to the initial costates can then be computed by Taylor expanding to first order the final states about the current trajectory with respect to the initial costates:

$$\chi_d(t_f,\alpha) = \chi_I(t_f,\alpha) + \frac{\partial \chi(t_f,\alpha)}{\partial \Lambda(t_o,\alpha)}\Big|_I [\Lambda_d(t_o,\alpha)-\Lambda_I(t_o,\alpha)] , \tag{4.3.73}$$

where

$$\alpha = [\alpha_1 \ \alpha_2]^T ,$$

$\chi_d(t_f,\alpha)$ is the desired modified final states, $\chi_I(t_f,\alpha)$ is the integrated final states of the current iterate, $\partial\chi(t_f,\alpha)/\partial\Lambda(t_o,\alpha)\big|_I$ is the partial derivative of the final states with respect to the initial costates computed about the trajectory of the current iteration, $\Lambda_d(t_o,\alpha)$ is the desired initial costate, and $\Lambda_I(t_o,\alpha)$ is the current initial costate. If the partial derivative matrix of (4.3.73) were non-singular, one could solve (4.3.73) for $\Lambda_d(t_o,\alpha)$ to produce the initial costate for the next iteration, as follows:

$$\Lambda_{I+1}(t_o,\alpha) = \Lambda_I(t_o,\alpha) - \left[\frac{\partial \chi(t_f,\alpha)}{\partial \Lambda(t_o,\alpha)}\right]^{-1}_I [\chi_I(t_f,\alpha)-\chi_d(t_f,\alpha)] , \tag{4.3.74}$$

where

$$\Lambda_{I+1}(t_o,\alpha) = \Lambda_d(t_o,\alpha) .$$

However, since the Euler parameter costates are not independent, the partial derivative matrix of (4.3.74) is rank deficient. To make the Euler parameter costates unique, the orthogonality constraint of (4.3.53) is applied to the new Euler parameter costates:

$$\beta^T(t_o,\alpha)\gamma_d(t_o,\alpha) = \beta^T(t_o,\alpha)[\gamma_I(t_o,\alpha) + \Delta\gamma_I] = 0 , \qquad (4.3.75)$$

where $\gamma_d(t_o,\alpha)$ is the desired initial Euler parameter costate, $\gamma_I(t_o,\alpha)$ is the initial Euler parameter costate of the current iterate, and $\Delta\gamma_I$ is the appropriate correction term. Since the orthogonality constraint is enforced at each iteration, we observe that $\gamma_I(t_o,\alpha)$ is orthogonal to $\beta(t_o,\alpha)$ in (4.3.75), which leads to

$$\beta^T(t_o,\alpha)\Delta\gamma_I = 0 ,$$

or

$$\beta^T(t_o,\alpha)[\gamma_d(t_o,\alpha) - \gamma_I(t_o,\alpha)] = 0 . \qquad (4.3.76)$$

Observing that $\gamma_d(t_o,\alpha)$ and $\gamma_I(t_o,\alpha)$ are partitions of $\Lambda_d(t_o,\alpha)$ and $\Lambda_I(t_o,\alpha)$, respectively, one can modify (4.3.73) to include the orthogonality constraint by replacing one of the first four rows of (4.3.73) by (4.3.76). For simplicity, let us choose the first row of (4.3.73). The modification consists of replacing the first elements of $\chi_d(t_f,\alpha)$ and $\chi_I(t_f,\alpha)$ by 0, and replacing the first row of $\partial\chi(t_f,\alpha)/\partial\Lambda(t_o,\alpha)$ by v^T, where

$$v = [\beta^T(t_o,\alpha) \quad 0^T]^T . \qquad (4.3.77)$$

The modified form of (4.3.73) can be written as

$$\bar{\chi}_d(t_f,\alpha) = \bar{\chi}_I(t_f,\alpha) + \bar{\Phi}[\Lambda_d(t_o,\alpha) - \Lambda_I(t_o,\alpha)] , \qquad (4.3.78)$$

where

$$\bar{\chi}_d(t_f,\alpha) = [0\ \beta_1\ \beta_2\ \beta_3\ \omega^T\ u_o^T\ u_1^T]_d^T \Big|_{t_f,\alpha} ,$$

$$\bar{\chi}_I(t_f,\alpha) = [0\ \beta_1\ \beta_2\ \beta_3\ \omega^T\ u_o^T\ u_1^T]_I^T \Big|_{t_f,\alpha} ,$$

$$\bar{\Phi} = \begin{bmatrix} \beta_o(t_o,\alpha) & \cdots & \beta_3(t_o,\alpha) & 0 & \cdots & 0 \\ \Phi_{\chi_2\Lambda_1} & & \cdots & & & \Phi_{\chi_2\Lambda_{13}} \\ \vdots & & & & & \vdots \\ \Phi_{\chi_{13}\Lambda_1} & & \cdots & & & \Phi_{\chi_{13}\Lambda_{13}} \end{bmatrix} ,$$

and

$$\Phi_{\chi_i\Lambda_j} = \frac{\partial\chi_i(t_f,\alpha)}{\partial\Lambda_j(t_o,\alpha)} \Big|_I .$$

To compute the initial costate for the next iteration, (4.3.78) may be solved for $\Lambda_d(t_o,\alpha)$, yielding

$$\Lambda_{I+1}(t_o,\alpha) = \Lambda_I(t_o,\alpha) - \bar{\Phi}^{-1}[\bar{\chi}_I(t_f,\alpha) - \bar{\chi}_d(t_f,\alpha)] , \qquad (4.3.79)$$

where

$$\Lambda_{I+1}(t_o,\alpha) = \Lambda_d(t_o,\alpha) .$$

The entire continuation process is summarized in algorithmic form as follows. (The single-axis diagonal inertia matrix starting guess of Section 4.3.4 is assumed to have been computed).

Step 1. If $\alpha_1 = 1$ and $\alpha_2 = 1$, stop. (end of continuation). Otherwise, increment α_1 and α_2, and compute $[I(\alpha_1)]$, $\chi(t_o,\alpha_2)$, and $\chi(t_f,\alpha_2)$.

Step 2. Integrate state-costate differential equations [(4.3.21) through (4.3.28)], and state-costate partials with respect to initial costates (Appendix I).

Step 3. Compute error in satisfaction of modified final boundary conditions. If small, go to Step 1.

Step 4. Compute new initial costates [(4.3.79)]. Go to Step 2.

4.3.6 Numerical Results

A 60 second rest-to-rest maneuver with angular displacements of 1 radian about each axis was simulated. The weighting matrix for the angular velocity was arbitrarily chosen as $Q = 10^{-3}I$, where I is the (3×3) identity matrix. In choosing the value for the break frequency, ω_B, it was found that it was best to choose ω_B so that it corresponded to the frequency of the maneuver, that is,

$$\omega_B = \frac{2\pi}{(t_f - t_o)} . \qquad (4.3.80)$$

For the above value of ω_B, the resulting maneuver had pseudo-controls with smooth profiles (Figure 4-4). For higher values of ω_B, the pseudo-control profiles of the resulting maneuver had more undulations (see Figure 4-5). This reflects the higher frequency content of the controls, directly resulting from the higher value of ω_B. For lower values of ω_B, the resulting trajectories were similar to the case where ω_B was chosen according to (4.3.80). However, the number of Newton-Raphson iterations

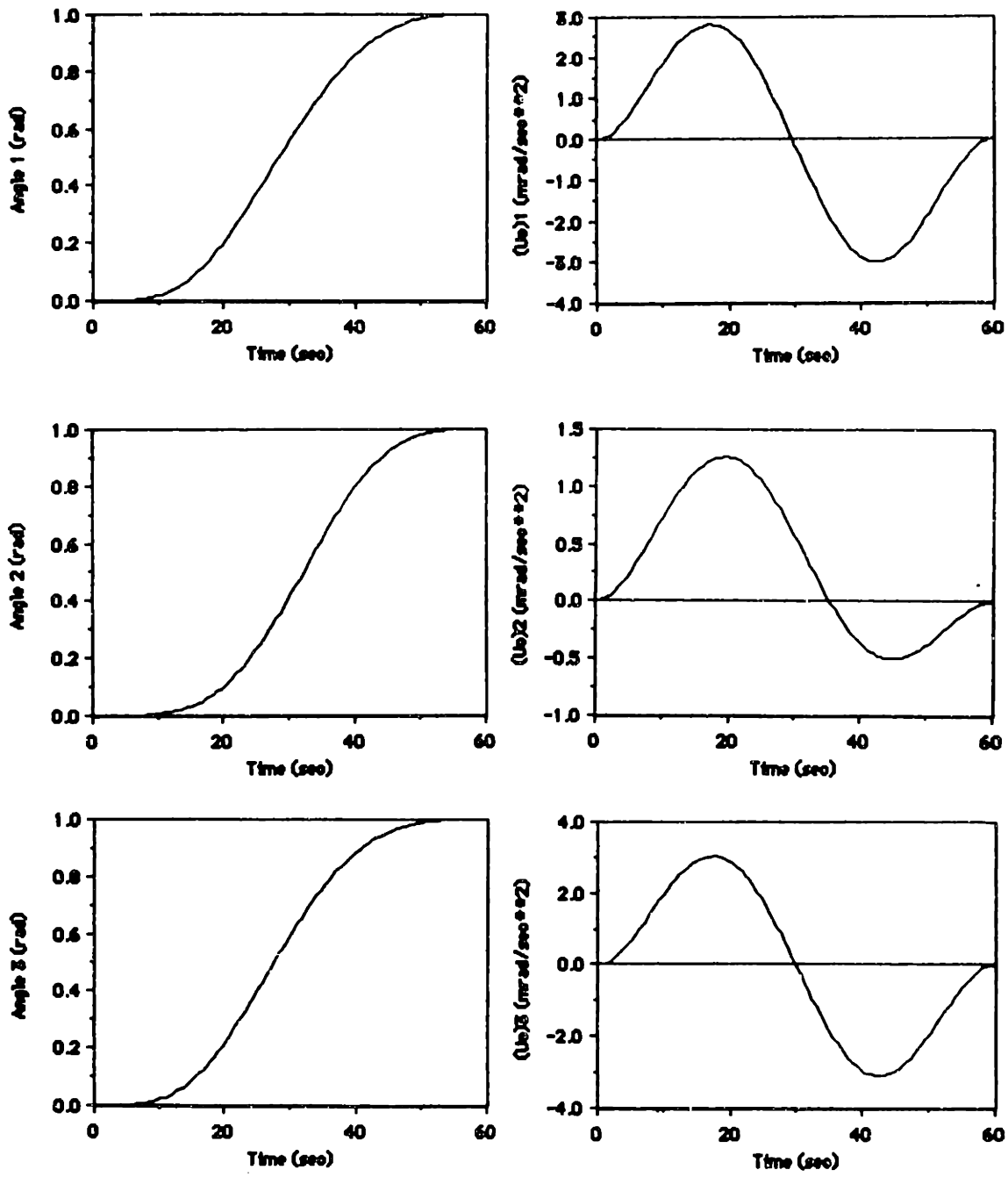


Figure 4-4. Rigid Body Angle and Pseudo-Control Profiles for $\omega_B = 2\pi/60 = 0.1047$ rad/s

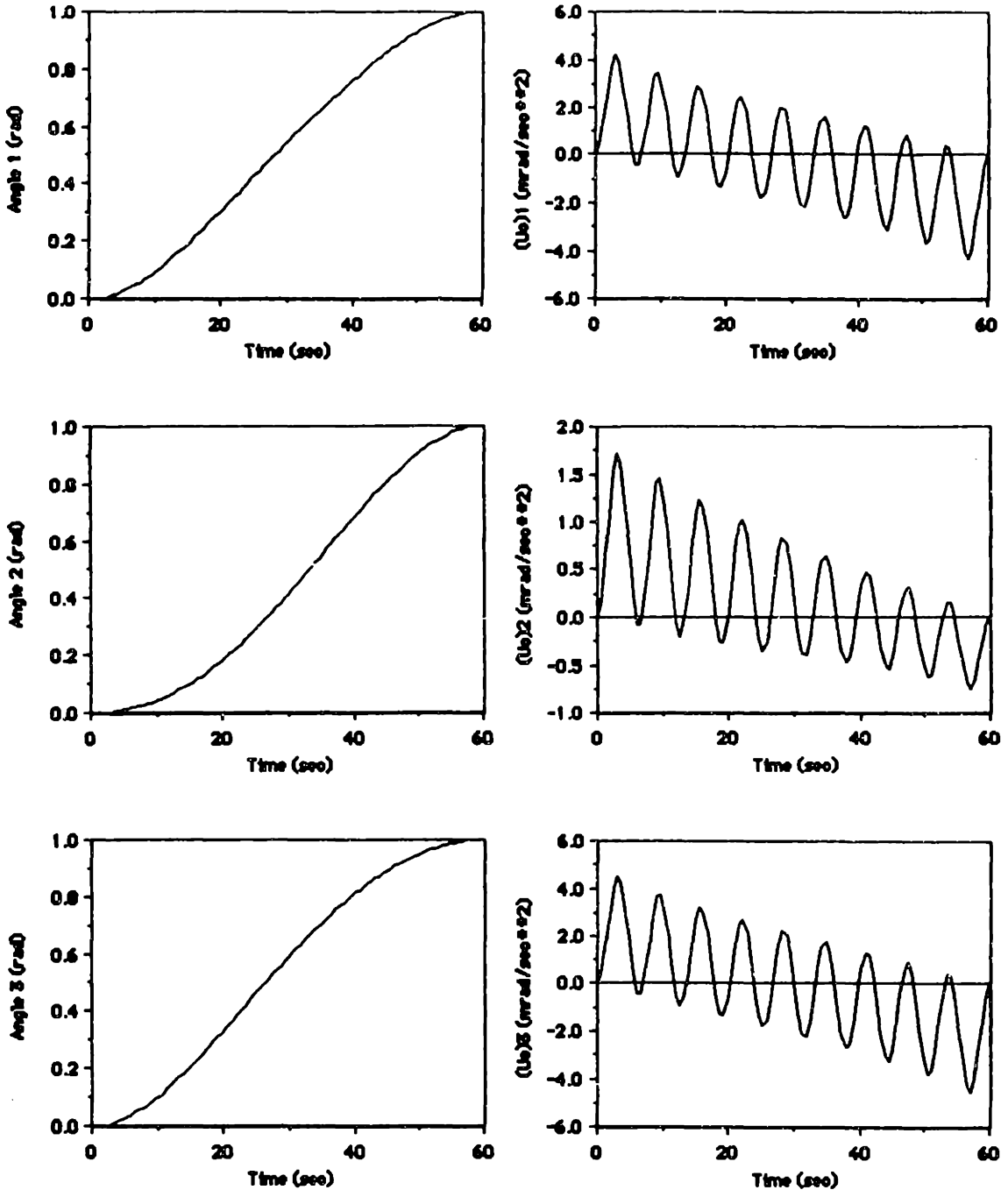


Figure 4-5. Rigid Body Angle and Pseudo-Control Profiles for ω_B = 1.0 rad/s

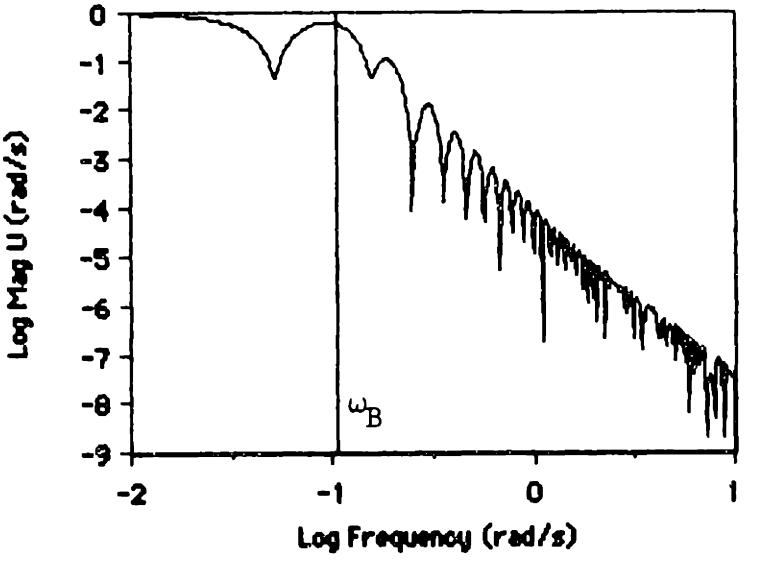
required for convergence was increased slightly, indicating that the partial derivative matrix of (4.3.79) may have become numerically stiffer. To illustrate the effect of the choice of ω_B on the frequency content of the resulting control, Figure 4-6 shows the frequency spectra of the pseudo-control for the single-axis starting guesses with ω_B corresponding to Figures 4-4, and 4-5. Because the penalty function of (4.3.19) penalizes the frequency content of the pseudo-controls for values of frequency above ω_B, one sees a sharp roll-off near ω_B in Figure 4-6.

For the example cases shown, the continuation parameter for the moments of inertia, α_1, was incremented while holding the continuation parameter for the boundary conditions, α_2, at 0. When $\alpha_1 = 1$, the value of α_2 was incremented to 1 while α_1 was held at 1. This is a more conservative approach, since this allows one to identify which quantity (moment of inertia or boundary condition) is the cause of numerical problems, should any occur. Consequently, if problems arise, then the step-size may be shortened for the continuation process for that quantity. In the example cases, the continuation was done in three steps for each quantity with about 3-6 Newton-Raphson iterations required at each increment. The convergence criterion for termination of the Newton-Raphson iteration was that the average number of unchanged digits was required to be 5 or more. For the final continuation step, this criterion was changed to require an average of 8 digits of accuracy. The continuation parameter increments were exponential, where the k-th continuation parameter is given by

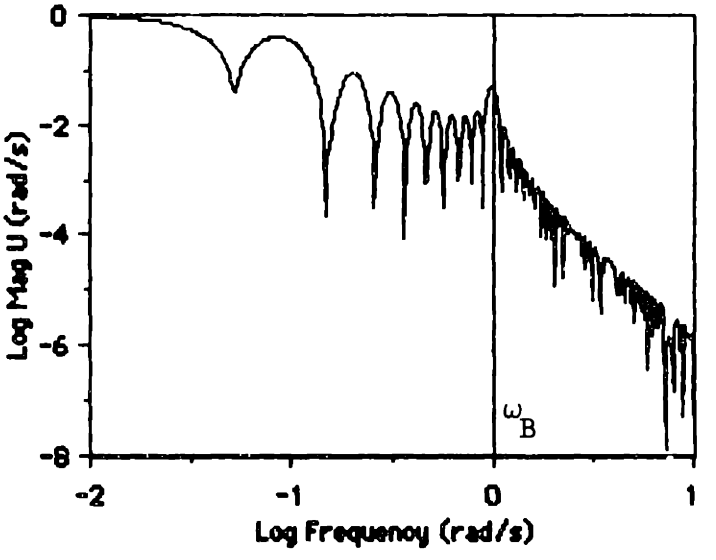
$$\alpha^{(k)} = e^{(k \ln 2)/n} - 1 , \qquad (4.3.81)$$

where $n = 3$ was the number of discrete steps. Thus, the entire continuation was performed with the following set of parameters: $(\alpha_1,\alpha_2) = (0,0)$, $(0.260,0)$, $(0.587,0)$, $(1,0)$, $(1,0.260)$, $(1,0.587)$, and $(1,1)$.

The second example case ($\omega_B = 1.0$ rad/s) was also run using a combined continuation process, with the following set of parameters:



a) ω_B = 0.1047 rad/s



b) ω_B = 1.0 rad/s

Figure 4-6. Frequency Spectra of the Pseudo-Controls for Starting Guesses with ω_B = 0.1047 rad/s and ω_B = 1.0 rad/s

(α_1,α_2) = (0,0), (0.260, 0.260), (0.587, 0.587), and (1,1). The first two continuation steps required 4 iterations each for convergence. The last step required 10 iterations to obtain an average of 8 digits of accuracy. Results were identical to the case where separate continuation processes were performed.

4.4 Perturbation Feedback for Controlling the Flexible Body Response

4.4.1 Plant Linearization and Gain Calculation

This section presents a perturbation feedback scheme for controlling the elastic deformations of a flexible body when subjected to the nominal rigid body torque profile of Section 4.3. The flexible plant dynamics is linearized about the rigid body nominal solution at several points in time. Steady-state feedback gains are computed based on these linearized plants and an infinite-time performance index with control-rate penalty.

The states used for the flexible plant model for the perturbation feedback are

$$x = [\omega_x\ \omega_y\ \omega_z\ \dot{\xi}_S^T\ \dot{\xi}_R^T\ \xi_S^T\ \xi_R^T\ \phi_x\ \phi_y\ \phi_z]^T , \qquad (4.4.1)$$

where ω_x, ω_y, and ω_z are the body angular velocities of the spacecraft bus, ξ_S and ξ_R are the modal amplitudes of the solar array and the radiometer, respectively, and ϕ_x, ϕ_y, and ϕ_z are angular displacements of the spacecraft bus. The modal amplitudes, ξ_S and ξ_R, represent a reduced order elastic model. The numerical simulation includes several additional elastic modes to represent the residual mode responses.

From DISCOS, the linearized system dynamics and control influence matrices are obtained numerically through a quadratic finite difference approximation. The linearized state differential equation is then

$$\dot{x}(t) = A^{(i)}x(t) + B^{(i)}u_o(t) , \qquad t = t_i , \tag{4.4.2}$$

where t_i is the instant in time at which the plant is linearized, $A^{(i)}$ and $B^{(i)}$ are the linearized state dynamics and control influence matrices at $t = t_i$, and $u_o(t)$ is the pseudo-control. The linearized perturbed system dynamics is then given by

$$\delta\dot{x}(t) = A^{(i)}\delta x(t) + B^{(i)}\delta u_o(t) , \qquad t = t_i . \tag{4.4.3}$$

For control-smoothing, the above state differential equations are augmented by the pseudo-control differential equations:

$$\delta\dot{u}_o = \delta u_1 , \tag{4.4.4}$$

and

$$\delta\dot{u}_1 = \delta u_2 . \tag{4.4.5}$$

The performance index used for computing each set of steady-state gains is

$$J = \frac{1}{2}\int_{t_o}^{\infty} [\delta x^T(t) \;\; \delta u_o^T(t) \;\; \delta u_1^T(t) \;\; \delta u_2^T(t)] \; W \begin{Bmatrix} \delta x(t) \\ \delta u_o(t) \\ \delta u_1(t) \\ \delta u_2(t) \end{Bmatrix} dt , \tag{4.4.6}$$

where

$$W = \begin{bmatrix} Q & 0 & 0 & 0 \\ 0 & I & 0 & I/\omega_B^2 \\ 0 & 0 & 0 & 0 \\ 0 & I/\omega_B^2 & 0 & I/\omega_B^4 \end{bmatrix} .$$

The penalties on the pseudo-controls, and their rates are in the form used in Gupta's frequency-shaped control smoothing (Appendix C). For each linearized plant, steady-state gains are computed based on the performance index of (4.4.6), and the dynamics equations of (4.4.3) through (4.4.5). Since the performance index is not rigorously minimized for the nonlinear plant, this feedback approach is suboptimal with respect to the performance index of (4.4.6).

For a given plant, described by $A^{(i)}$ and $B^{(i)}$, the optimal steady-state feedback solution for minimizing the performance index of (4.4.6) is given by

$$\delta u_2(t) = - R^{-1}[\tilde{B}^T P_{ss}^{(i)} + N^T]\delta\bar{\chi}(t) , \qquad (4.4.7)$$

where

$$R = I/\omega_B^4 , \qquad N^T = [0 \quad I/\omega_B^2 \quad 0] ,$$

$$\tilde{Q} = \begin{bmatrix} Q & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} , \qquad W = \begin{bmatrix} \tilde{Q} & N \\ N^T & R \end{bmatrix} ,$$

$$\tilde{A} = \begin{bmatrix} A^{(i)} & B^{(i)} & 0 \\ 0 & 0 & I \\ 0 & 0 & 0 \end{bmatrix} , \qquad \tilde{B}^T = [0 \quad 0 \quad I] ,$$

$$0 = P_{ss}^{(i)}\tilde{A} + \tilde{A}^T P_{ss}^{(i)} - [P_{ss}^{(i)}\tilde{B} + N]R^{-1}[N^T + \tilde{B}^T P_{ss}^{(i)}] + \tilde{Q} ,$$

and

$$\delta\bar{\chi}(t) = [\delta x^T(t) \quad \delta u_o^T(t) \quad \delta u_1^T(t)]^T .$$

During the perturbation feedback, the feedback gains are interpolated between the points in time at which the gains are computed.

That is, denoting the feedback gains by F, so that

$$\delta u_2(t) = - F(t)\delta\tilde{\chi}(t) \ , \tag{4.4.8}$$

one can interpolate the feedback gains by

$$F(t) = \left(\frac{t-t_i}{t_{i+1}-t_i}\right)F^{(i+1)} + \left(\frac{t_{i+1}-t}{t_{i+1}-t_i}\right)F^{(i)} \ , \tag{4.4.9}$$

where

$$F^{(i)} = R^{-1}[\tilde{B}^T P_{ss}^{(i)} + N^T] \ ,$$

and $t_i \leq t \leq t_{i+1}$. For the nominal rigid body state, the modal amplitude portions of $\tilde{\chi}(t)$ are identically zero.

4.4.2 Numerical Results

Example cases were generated with the assumption of perfect state estimation. (Section 4.5 shows example cases where the Kalman filter is used for state estimation.) The 60 second rest-to-rest maneuver with 1 radian angular displacement about each axis discussed in Section 4.3.6 is used here. The flexible plant was linearized about the rigid body nominal solution at 12 second intervals. Several off-nominal cases were studied. For all cases, the two lowest solar array modes and the two lowest radiometer modes were chosen for inclusion in the feedback formulation. The other higher frequency modes represent residual modes. All modes are assumed to have 0.1% damping. The break frequency for the perturbation controller, ω_B, was chosen to be half the frequency of the highest controlled mode, so as to minimize the excitation of the residual modes. Figure 4-7 shows the frequency spectra of the pseudo-control corrections when the perturbation feedback controller is subjected to initial conditions.

Case 1 (Figure 4-8) is the 'nominal' flexible body case, with

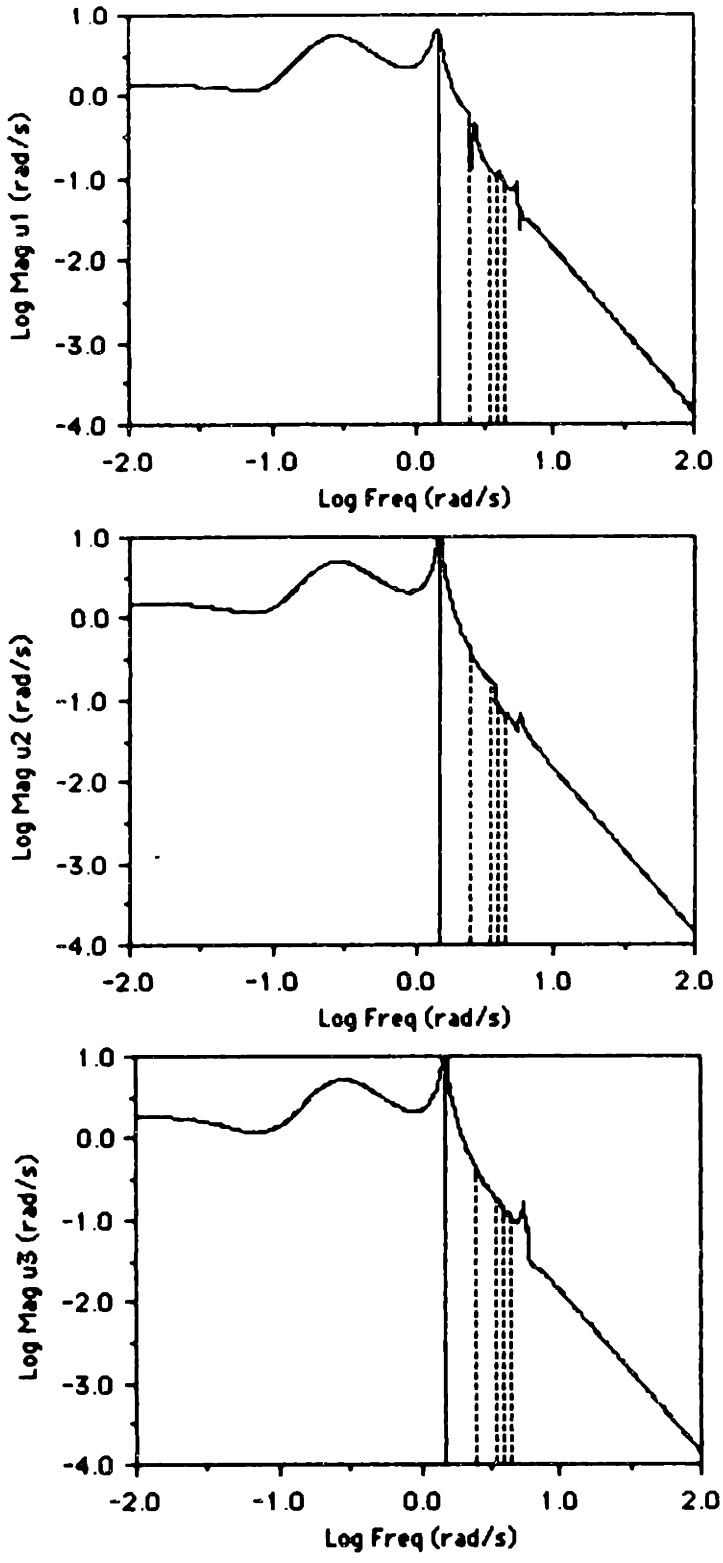


Figure 4-7. Typical Frequency Spectra of the Pseudo-Control Corrections

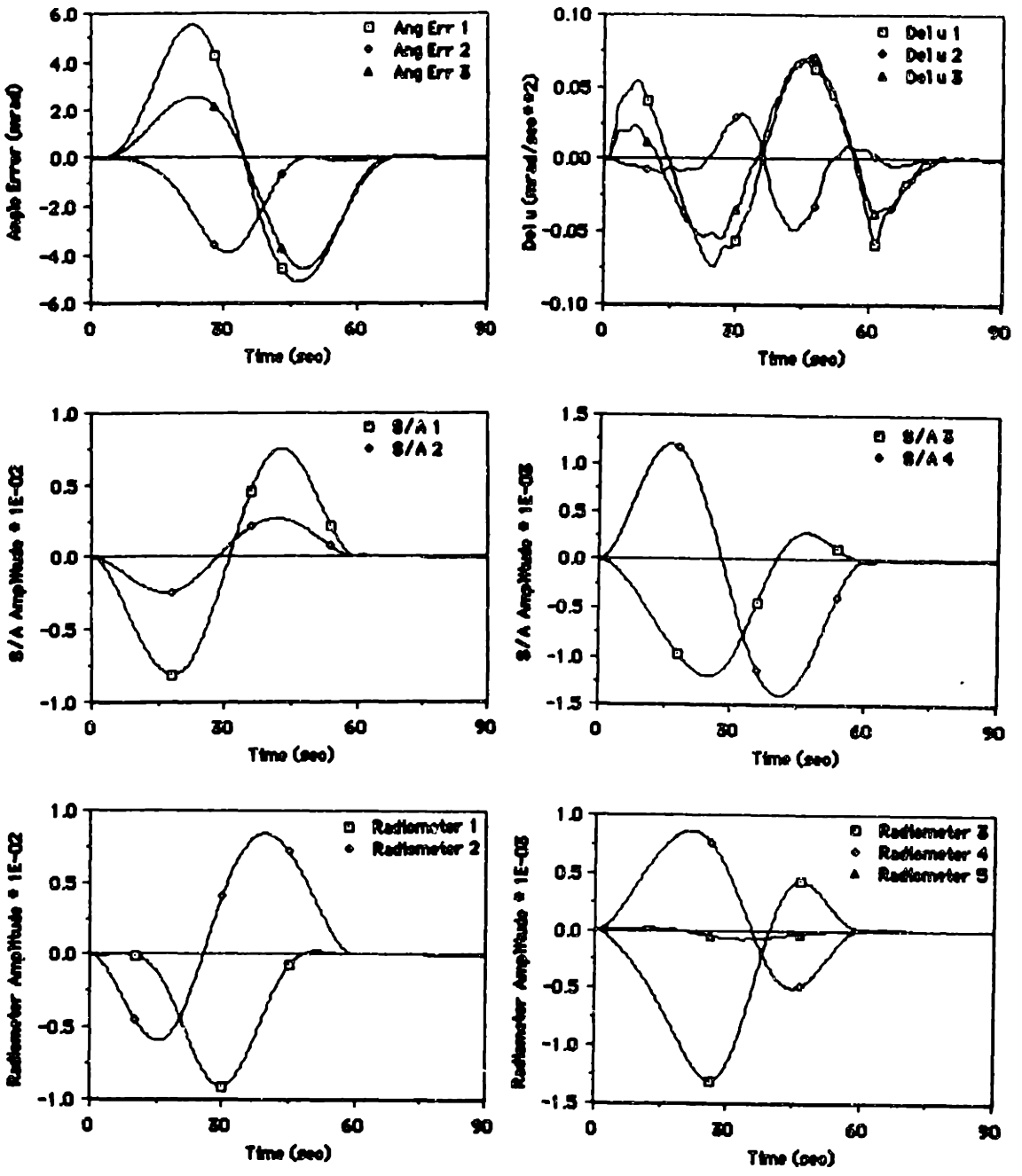


Figure 4-8. Case 1. 'Nominal' Perturbation Feedback

perfect plant knowledge and nominal initial conditions. The controlled modal amplitudes and residual modal amplitudes for the solar array (denoted as S/A) and the radiometer are plotted separately. Note that all the modal amplitudes are very small by the end of the maneuver. The angle errors take a slightly longer time to settle. The pseudo-control corrections (denoted by Del u), take about 20 seconds beyond the maneuver time to damp out.

Case 2 (Figures 4-9 and 4-10) is the same as Case 1, except that all the modal amplitudes, including the residual modes, are given non-zero initial values. The initial amplitudes are chosen so as to be approximately half the peak values experienced in Case 1. The controlled modes for both the solar array and radiometer exhibit higher damping than the residual modes. The general shapes of the modal responses are quite close to those shown for Case 1, except that in Case 2 there are higher frequency vibrations superimposed over the smooth modal response curves. The effect of the non-zero initial modal amplitudes on the pseudo-control corrections is apparent in the higher frequency content of the time-history plots. Note that the overall frequency of the undulations in the pseudo-control correction plots is much lower than for the modal amplitude plots, thanks to the control smoothing formulation. Note also that after about 30 seconds into the maneuver, the pseudo-control correction plots have almost the same shape as those shown in Case 1.

Case 3 (Figures 4-11 and 4-12) is the same as Case 1, except that initial angular errors are specified for Case 3. The error in the initial angle is chosen to be 5% of the total angular displacement about each Euler axis. The sign of each of the errors is arbitrarily assigned. Note that the initial angular errors are an order of magnitude higher than the peak angular errors shown in Case 1. After about 20 seconds, the angular error and modal amplitude time histories approach the general shapes of the corresponding plots for Case 1. Since the peak values for the pseudo-control corrections are more than an order of magnitude higher than those for Case 1, the peak modal amplitudes are also higher than in Case 1. It appears that the oscillations in the pseudo-control corrections near the initial time may have excited the third radiometer mode (a residual mode).

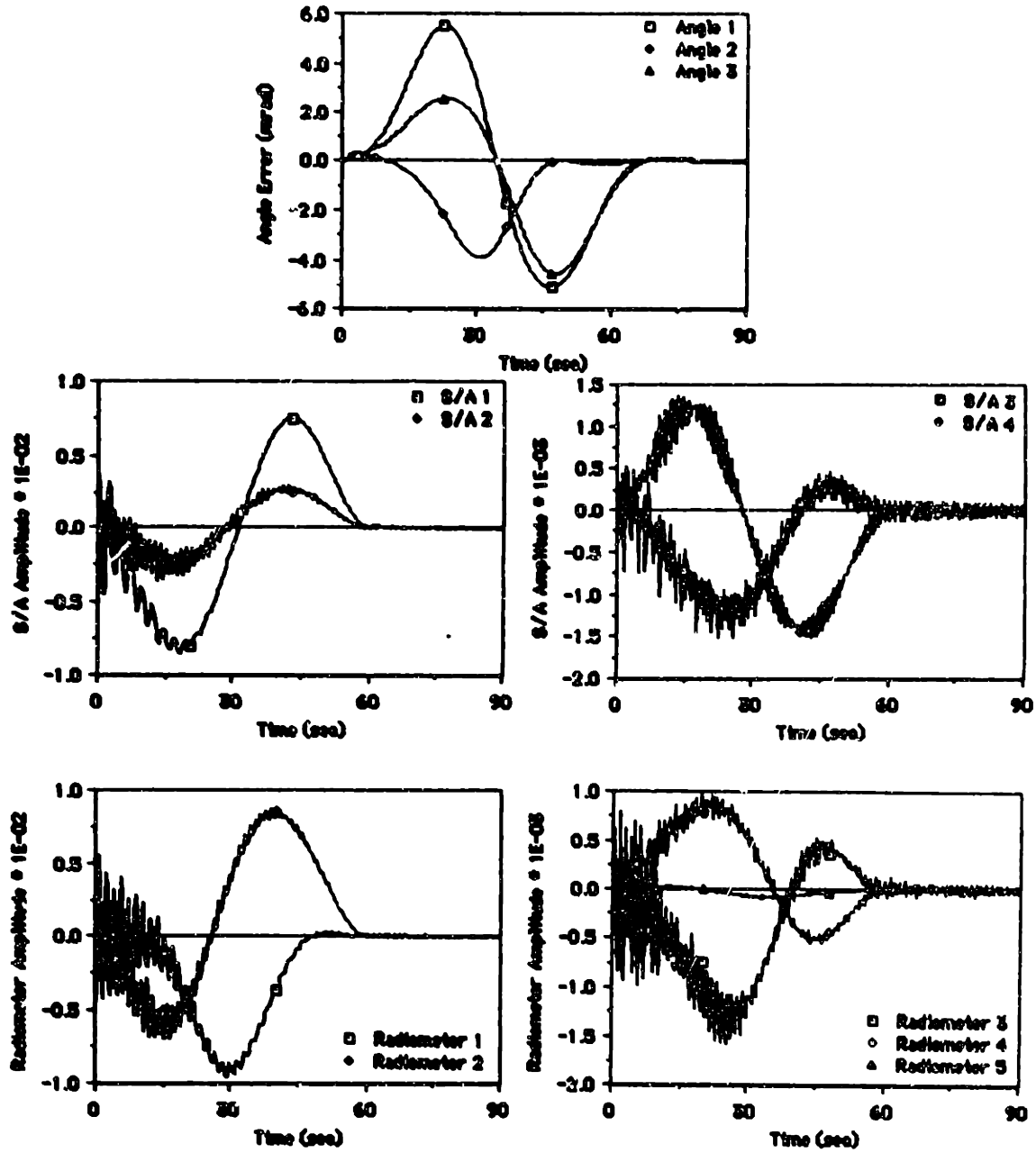


Figure 4-9. Case 2. Non-Zero Initial Amplitudes on All Modes

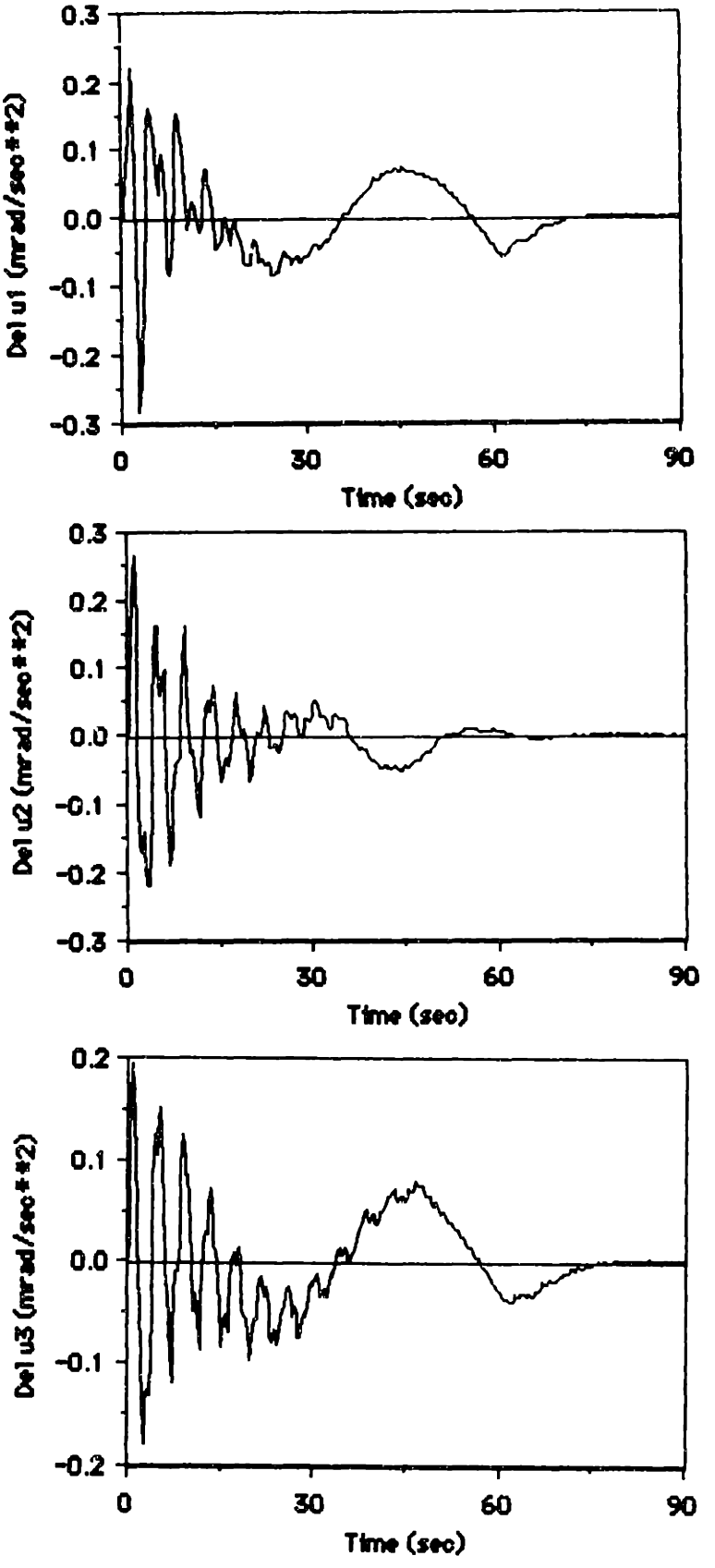


Figure 4-10. Pseudo-Control Corrections for Case 2

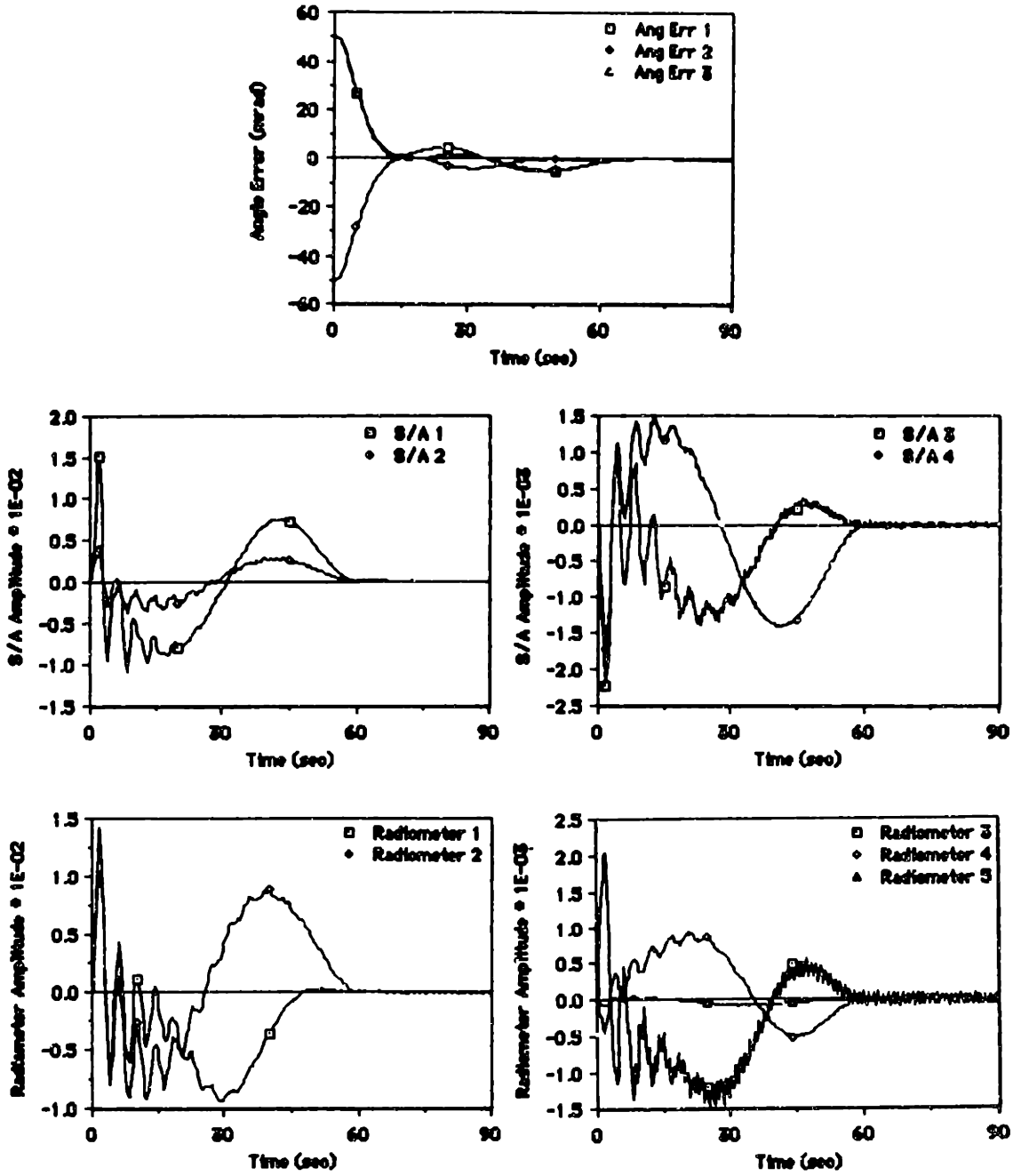


Figure 4-11. Case 3. Off-Nominal Initial Angles

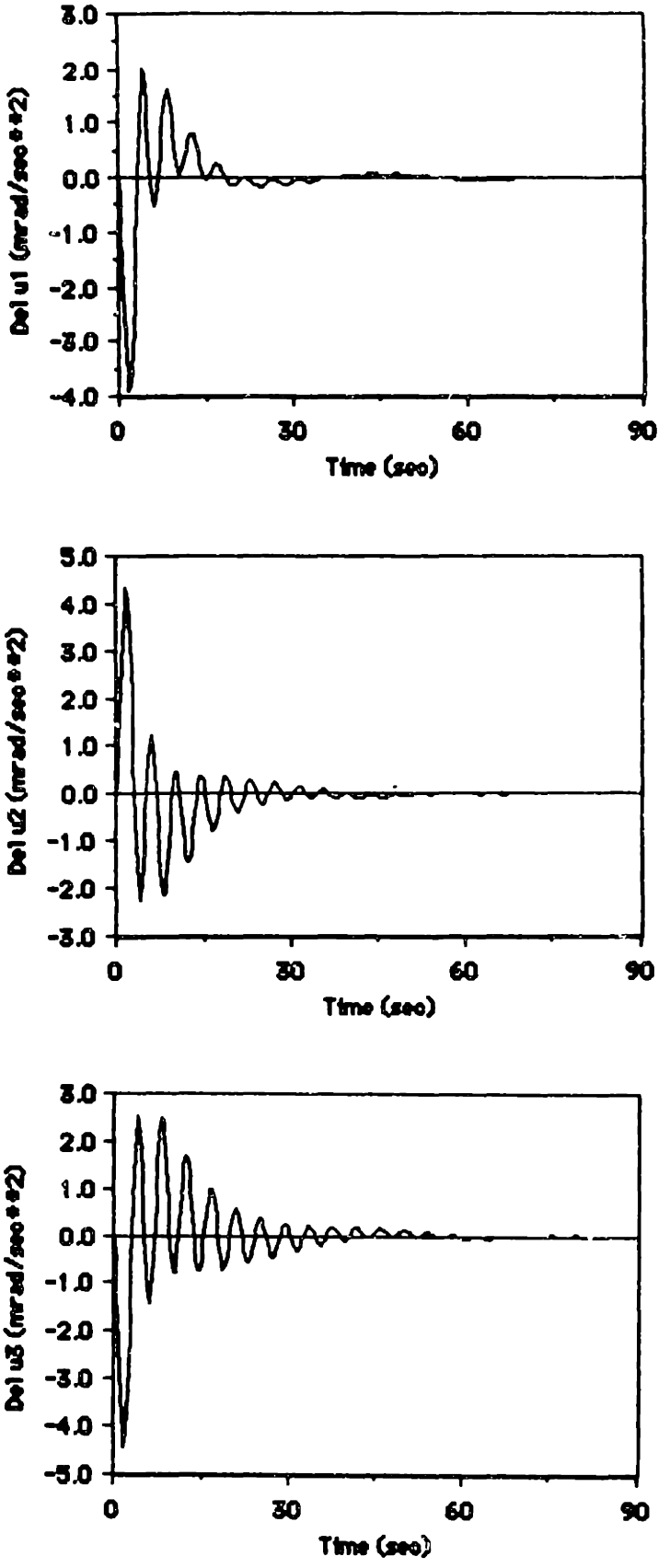


Figure 4-12. Pseudo-Control Corrections for Case 3

Case 4 (Figure 4-13) is the same as Case 1, except that the moments of inertia of the spacecraft bus are altered by 10%. The nominal inertia matrix for the spacecraft bus is given in slug-ft^2 by

$$[I] = \begin{bmatrix} 6247 & -13.61 & -113.6 \\ -13.61 & 6247 & 326.7 \\ -113.6 & 326.7 & 609.9 \end{bmatrix}, \qquad (4.4.10)$$

and the off-nominal inertia matrix for Case 4 is

$$[I] = \begin{bmatrix} 6867 & -15.00 & -102.6 \\ -15.00 & 5627 & 359.7 \\ -102.6 & 359.7 & 670.9 \end{bmatrix}. \qquad (4.4.11)$$

The mass properties of the solar array and radiometer remain unaltered. The solar array and radiometer modal amplitude time-histories resemble those for Case 1. However, the angular error and pseudo-control correction time-histories are much affected by the off-nominal mass properties. In particular, the peak pseudo-control corrections are several times higher than in Case 1. Nevertheless, the perturbation feedback controller is able to control the spacecraft to its desired final orientation and maintain it there after the end of the maneuver.

4.5 Kalman Filter for Observing the System States

4.5.1 Gain Calculation

This section presents a modified Kalman filter to estimate the system states used in the perturbation feedback scheme of Section 4.4. The approach presented here involves the use of linearized plant equations similar to those used for the perturbation feedback.

Let us assume that in addition to the system dynamics matrix, $A^{(i)}$, and the control influence matrix, $B^{(i)}$, of (4.4.2), the measurement influence matrix, $C^{(i)}$, is also linearized about the nominal trajectory at

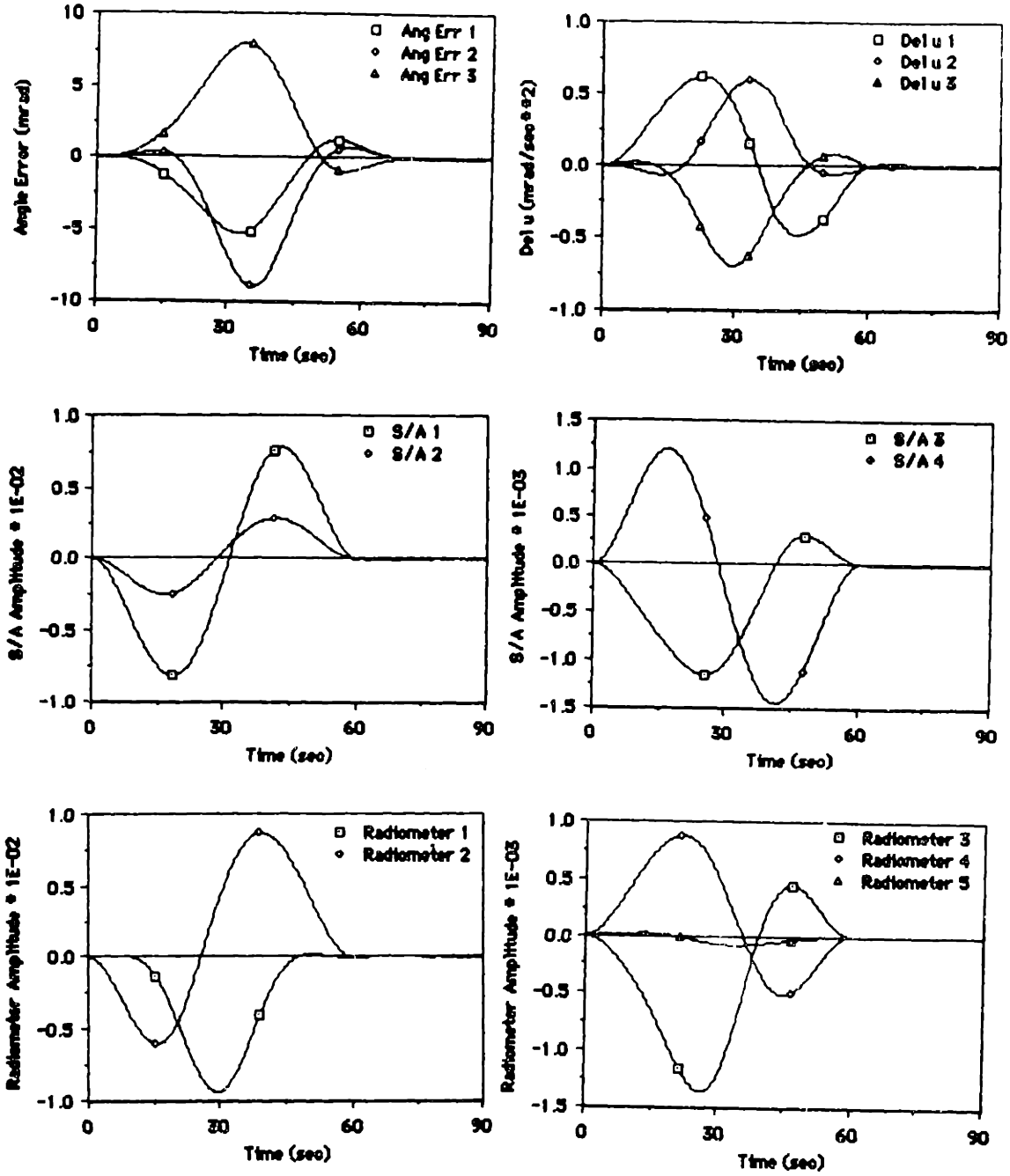


Figure 4-13. Case 4. Off-Nominal Inertia Matrix

several points in time. For the calculation of the Kalman gains, let us assume that the linearized plant dynamics and measurements are subjected to Gaussian white noise disturbances:

$$\dot{x}(t) = A^{(i)}x(t) + B^{(i)}u_o(t) + w(t) , \qquad t = t_1 , \tag{4.5.1}$$

and

$$y(t) = C^{(i)}x(t) + v(t) , \tag{4.5.2}$$

where

$$E\left[\begin{Bmatrix} w(t_1) \\ v(t_1) \end{Bmatrix}\begin{Bmatrix} w(t_2) \\ v(t_2) \end{Bmatrix}^T\right] = \begin{bmatrix} \bar{Q} & 0 \\ 0 & \bar{R} \end{bmatrix}\delta(t_2-t_1) ,$$

$$E[w(t)] = 0 , \quad \text{and } E[v(t)] = 0 .$$

Let us assume a linear estimator of the form

$$\dot{\hat{x}}(t) = A^{(i)}\hat{x}(t) + B^{(i)}u_o(t) + K^{(i)}(t)[y(t) - C^{(i)}\hat{x}(t)] , \tag{4.5.3}$$

where $K^{(i)}(t)$ is a set of observer gains, and let us choose the gain matrix, $K^{(i)}$, so that the error covariance is minimized. The estimation error may be defined as

$$e(t) = x(t) - \hat{x}(t) , \tag{4.5.4}$$

with its differential equation given by

$$\dot{e}(t) = [A - K(t)C]e(t) + w(t) - K(t)v(t) , \tag{4.5.5}$$

from (4.5.1) and (4.5.3), and we have dropped the superscript (i). Using Itô calculus, we can write the solution to (4.5.5) in the form [44,46]:

$$e(t) = \Phi(t,t_o)e(t_o) + \int_{t_o}^{t} \Phi(t,\tau)d\beta_\tau - \int_{t_o}^{t} \Phi(t,\tau)K(\tau)d\gamma_\tau \ , \qquad (4.5.6)$$

where

$$\dot{\Phi}(t,t_o) = [A - K(t)C]\Phi(t,t_o) \ , \quad \Phi(t_o,t_o) = I \ ,$$

$$d\beta_\tau \triangleq \beta(\tau+\Delta\tau) - \beta(\tau) \ , \quad \Delta\tau \to 0 \ ,$$

$$d\gamma_\tau \triangleq \gamma(\tau+\Delta\tau) - \gamma(\tau) \ , \quad \Delta\tau \to 0 \ ,$$

and $\beta(t)$ and $\gamma(t)$ are the Wiener processes corresponding to the white noise $w(t)$ and $v(t)$, respectively. The error covariance matrix is then given by

$$X(t) \triangleq E[e(t)e^T(t)]$$

$$= \Phi(t,t_o)X(t_o)\Phi^T(t,t_o) + \int_{t_o}^{t} \Phi(t,\tau)\bar{Q}\Phi^T(t,\tau)d\tau$$

$$+ \int_{t_o}^{t} \Phi(t,\tau)K(\tau)\bar{R}K^T(\tau)\Phi^T(t,\tau)d\tau \ , \qquad (4.5.7)$$

where we have used the fact that

$$E[\int_{t_o}^{t} g(\tau)d\beta_\tau] = 0 \ ,$$

$$E[\int_{t_o}^{t} g(\tau)d\gamma_\tau] = 0 \ ,$$

$$E[d\beta_\tau (d\gamma_\tau)^T] = 0 ,$$

$$E[d\beta_\tau (d\beta_\tau)^T] = \bar{Q} d\tau ,$$

and

$$E[d\gamma_\tau (d\gamma_\tau)^T] = \bar{R} d\tau ,$$

where g(t) is any deterministic function or independent stochastic function. Differentiation of (4.5.7) leads to

$$\dot{X}(t) = [A - K(t)C]X(t) + X(t)[A - K(t)C]^T + \bar{Q} + K(t)\bar{R}K^T(t)$$

$$= AX(t) + X(t)A^T - X(t)C^T\bar{R}^{-1}CX(t) + \bar{Q}$$

$$+ [K(t) - X(t)C^T\bar{R}^{-1}]\bar{R}[K(t) - X(t)C^T\bar{R}^{-1}]^T , \qquad (4.5.8)$$

which is the covariance propagation equation. Since the last term of the second equation in (4.5.8) is quadratic, and $\bar{R}$ is positive definite, one can see that the error covariance is minimized by choosing

$$K(t) = X(t)C^T\bar{R}^{-1} , \qquad (4.5.9)$$

which is the Kalman gain matrix [14,53,58].

For the Kalman filter approach used in this section, we compute steady state Kalman gains for each of the linearized plants $A^{(i)}$ and $C^{(i)}$. That is, we use

$$K^{(i)} = X_{ss}^{(i)}[C^{(i)}]^T\bar{R}^{-1} , \qquad (4.5.10)$$

where $X_{ss}^{(i)}$ is the steady-state error covariance matrix for the linearized plant, obtained by substituting (4.5.9) into (4.5.8), and setting $\dot{X}(t)$ to zero:

$$0 = AX_{ss} + X_{ss}A^T - X_{ss}C^T\bar{R}^{-1}CX_{ss} + \bar{Q} . \qquad (4.5.11)$$

During the simulation of the estimator, the Kalman gains are linearly interpolated as in Section 4.4 for the perturbation feedback gains. That is,

$$K(t) = \left(\frac{t-t_i}{t_{i+1}-t_i}\right)K^{(i+1)} + \left(\frac{t_{i+1}-t}{t_{i+1}-t_i}\right)K^{(i)} , \qquad (4.5.12)$$

where $t_i \leq t \leq t_{i+1}$, and $K^{(i)}$ and $K^{(i+1)}$ are the steady-state Kalman gains computed at t_i and t_{i+1}, respectively.

4.5.2 Numerical Results

As in Section 4.4.2, the flexible plant was linearized about the rigid body nominal solution at 12 second intervals for the results of this section. The measurement variables used for the results of this section are the spacecraft Euler angles relative to the inertial frame; the body angular velocities of the spacecraft bus; the out-of-plane deformations and velocities of diagonally opposite corners of the solar array; and the deformations and velocities of two points on the radiometer. It is assumed that raw data from sensors, such as accelerometers on the solar array, has already been processed to provide the measurements stated above. Due to the limited scope of this study, the sensor locations were not optimized for best performance. The process and measurement noise variances were chosen as small percentages of the peak values experienced in the 'nominal' Case 1 of Section 4.4.2. The process noise w(t) may be partitioned in the same manner as the state vector of (4.4.1):

$$w(t) = [w_\omega^T \quad w_{\dot{S}}^T \quad w_{\dot{R}}^T \quad w_S^T \quad w_R^T \quad w_\phi^T]^T ,$$

where the subscripts ω and ϕ denote the process noise for the body angular velocities and Euler angles, respectively; and the subscripts S, R, $\dot{S}$ and $\dot{R}$ denote the process noise for the solar array and radiometer modal amplitudes, and their derivatives, respectively. The measurement noise v(t) may be partitioned as:

$$v(t) = [v_\omega^T \quad v_\phi^T \quad v_S^T \quad v_R^T]^T ,$$

where v_ω and v_ϕ represent the measurement noise for the body angular velocities and Euler angles, respectively; and v_S and v_R represent the measurement noise for solar array and radiometer measurements, respectively. The solar array measurements include absolute velocities (S1 and S2) and elastic deformations (S3 and S4) of two diagonally opposite corners (sensor points 3 and 6) of the solar array in the z direction, which is the direction for out-of-plane motion of the solar array. The radiometer measurements include absolute velocities (R1 and R2) and elastic deformations (R3 and R4) of two points, sensor points 9 and 10, near the center of the radiometer. The measurement at sensor point 9 is in the z direction, and the measurement for sensor point 10 is in the y direction. These measurements at the two sensor points were heuristically chosen among five convenient points (e.g. furthest point on the rim, elbow of the support boom) so that the measurement influence coefficients for the residual modes were smaller than those for the controlled/observed modes. For the above choice of measurements for the radiometer, the largest measurement influence coefficient for the residual modes was about half the magnitude of the coefficient for the first mode. Thus, the estimate for the amplitude of the first radiometer mode may be corrupted by observation spillover. (One may select optimal locations for the sensor points. However, this is beyond the scope of the present study.) Table 4-4 shows the noise variances used for computing the steady-state Kalman gains of each of the linearized plants.

Case 5 (Figures 4-14 through 4-16) shows the result of replacing

Table 4-4. Noise Variances for Computing Steady-State Kalman Gains

Process Noise

$\sigma^2(w_\omega)$ = diag [1.5(-4), 5.0(-5), 1.5(-4)]

$\sigma^2(w_{\dot{S}})$ = diag [1.7(-5), 5.6(-6)]

$\sigma^2(w_{\dot{R}})$ = diag [1.5(-5), 1.6(-5)]

$\sigma^2(w_S)$ = diag [1.0(-4), 3.0(-5)]

$\sigma^2(w_R)$ = diag [7.0(-5), 1.0(-4)]

$\sigma^2(w_\phi)$ = diag [5.0(-3), 1.5(-3), 5.0(-3)]

Measurement Noise

$\sigma^2(v_\omega)$ = diag [2.0(-2), 7.0(-3), 2.0(-2)]

$\sigma^2(v_\phi)$ = diag [1.0(-4), 1.0(-4), 1.0(-4)]

$\sigma^2(v_S)$ = diag [2.0(-4), 2.0(-4), 3.0(-3), 3.0(-3)]

$\sigma^2(v_R)$ = diag [3.0(-4), 4.0(-4), 3.0(-3), 3.0(-3)]

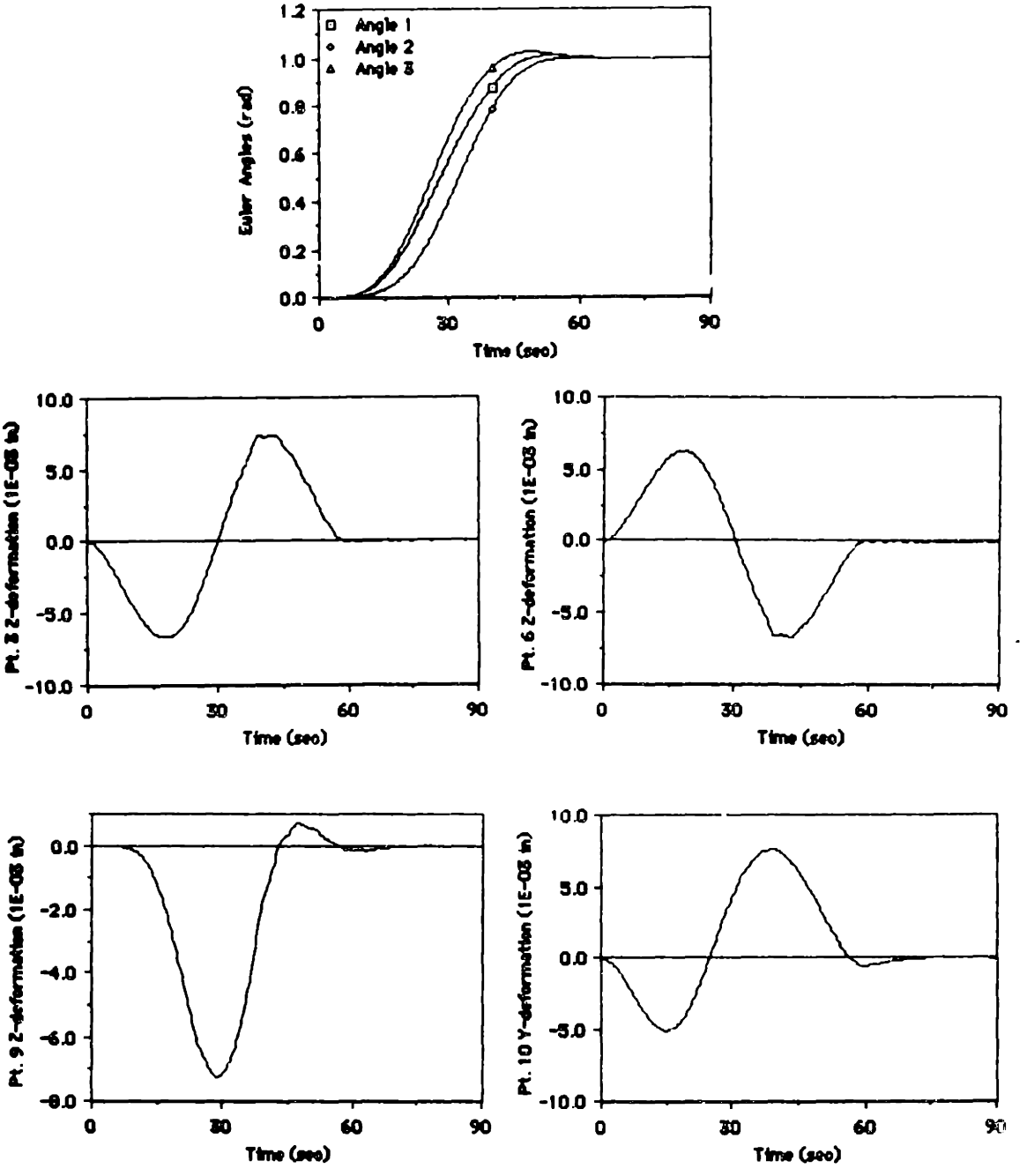


Figure 4-14. Euler Angles and Sensor Point Deformations For Case 5

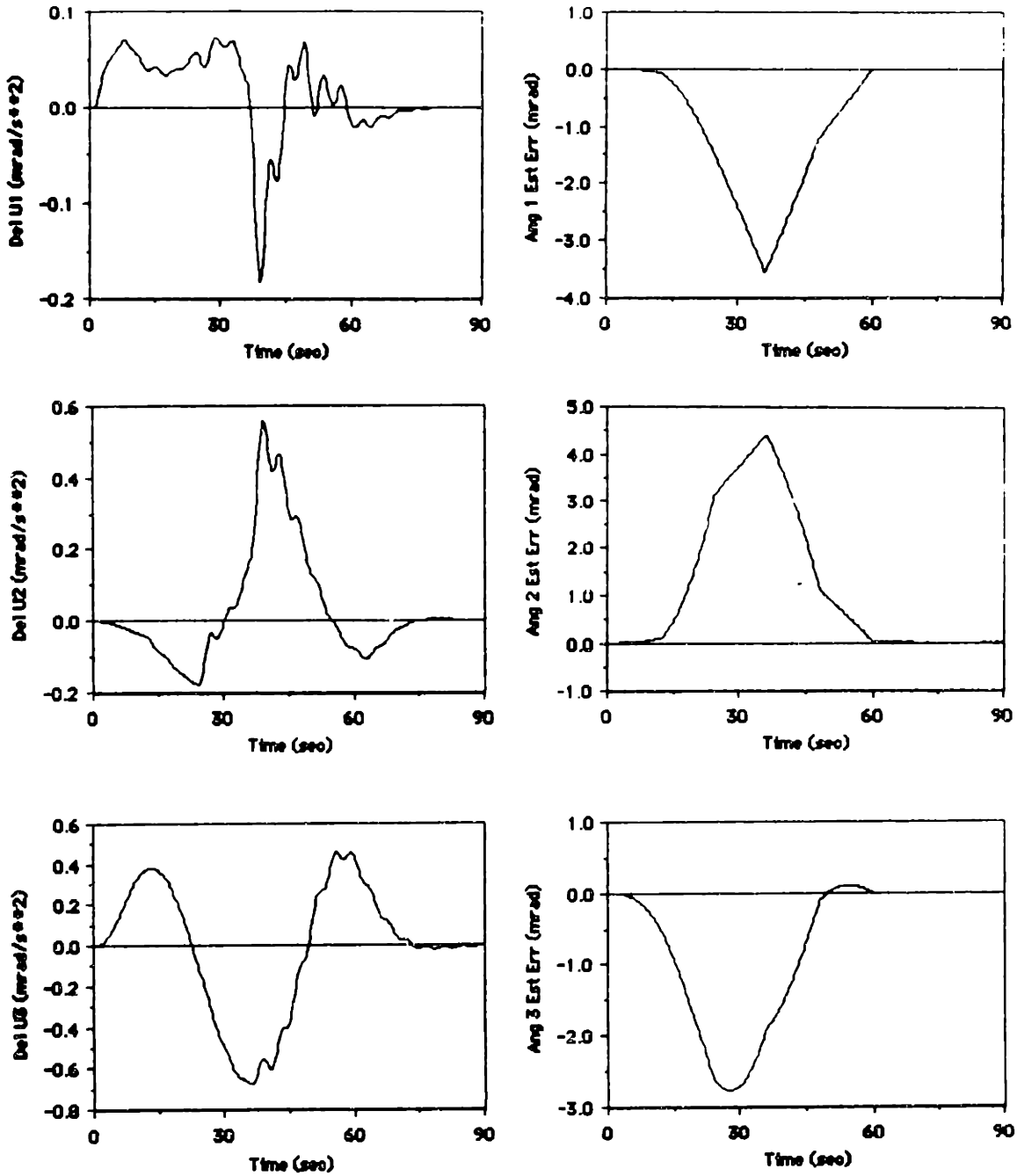


Figure 4-15. Pseudo-Control Corrections and Angle Estimate Errors for Case 5

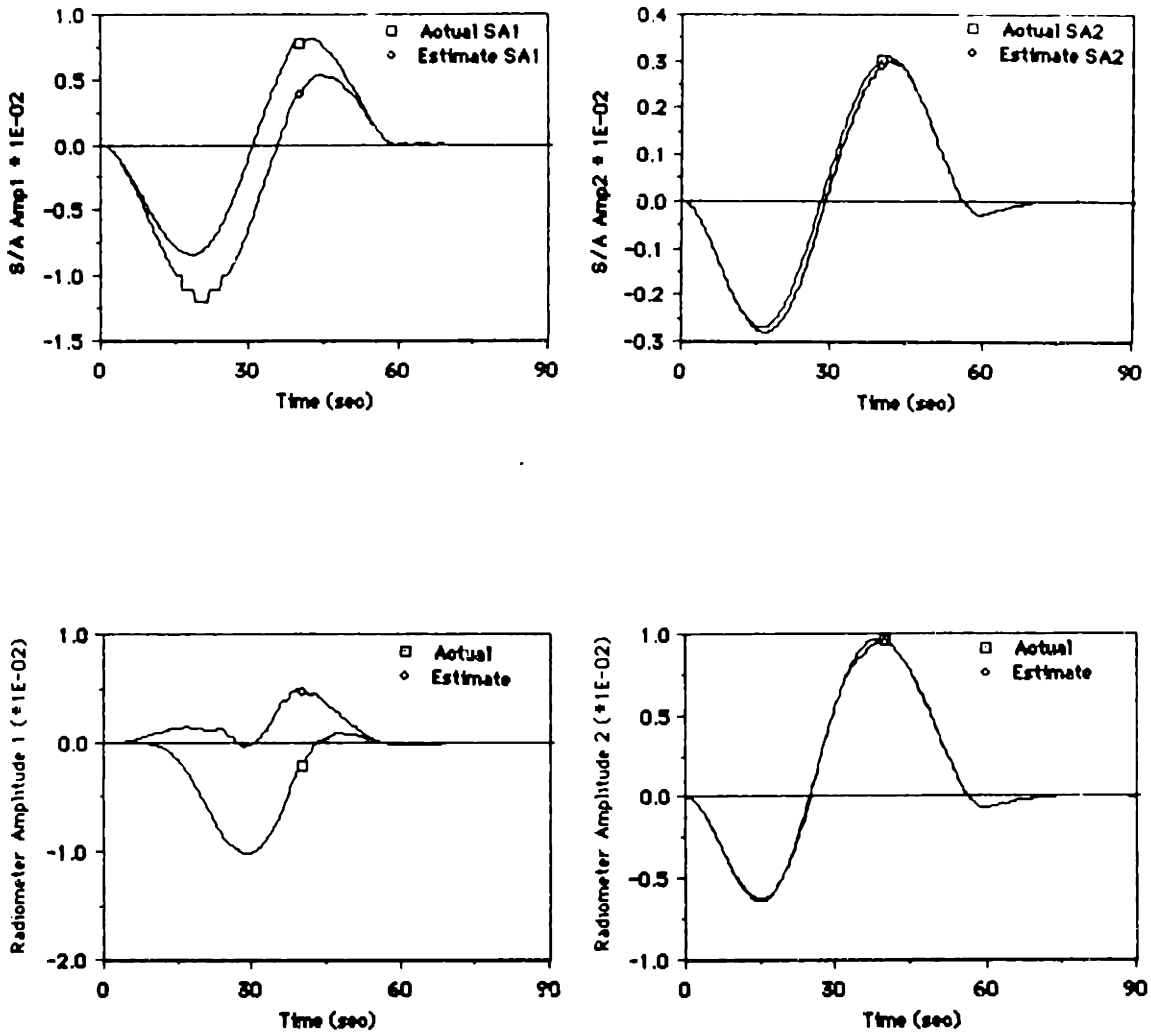


Figure 4-16. Solar Array and Radiometer Modal Amplitudes and Their Estimates for Case 5

the true state variable by the state estimate in computing the perturbation feedback for Case 1 of Section 4.4. Figure 4-14 shows that the Euler angles converge to their desired final values of 1 radian, with slight overshoot for the first and third Euler angles. The sensor point deformations shown in Figure 4-14 have very smooth profiles which converge to zero near the final maneuver time. Points 3 and 6 correspond to the two corners of the solar array, and points 9 and 10 correspond to two points near the center of the radiometer. The pseudo-control corrections of Figure 4-15 have higher peak values than the corresponding plots in Case 1 of Section 4.4. The angular estimate errors show the result of linearization at discrete points in time. One remedy is to linearize the plant at shorter time intervals. A better solution is to perform perturbation estimation about the nominal rigid body trajectory rather than estimation for the entire state. Figure 4-16 shows the amplitudes of the controlled modes and their estimates. One can see that the first solar array and radiometer modes are not estimated very well. This is due to observation spillover from the residual modes. In order to minimize the effects of observation spillover, one must choose optimal locations for the sensor points. Since there are 271 grid points on the radiometer with 5 modes, and 1000 grid points on the solar array with 4 modes (the original model has 15 modes), an automated procedure must be used for the selection of sensor locations [39,107]. However, this is beyond the scope of the current study.

Case 6 (Figures 4-17 through 4-19) is the same as Case 5, except that the angular accelerations are subjected to zero mean process noise, with variances given by

$$\sigma^2(w_1) = 1.5 \times 10^{-5} ,$$

$$\sigma^2(w_2) = 5.0 \times 10^{-6} ,$$

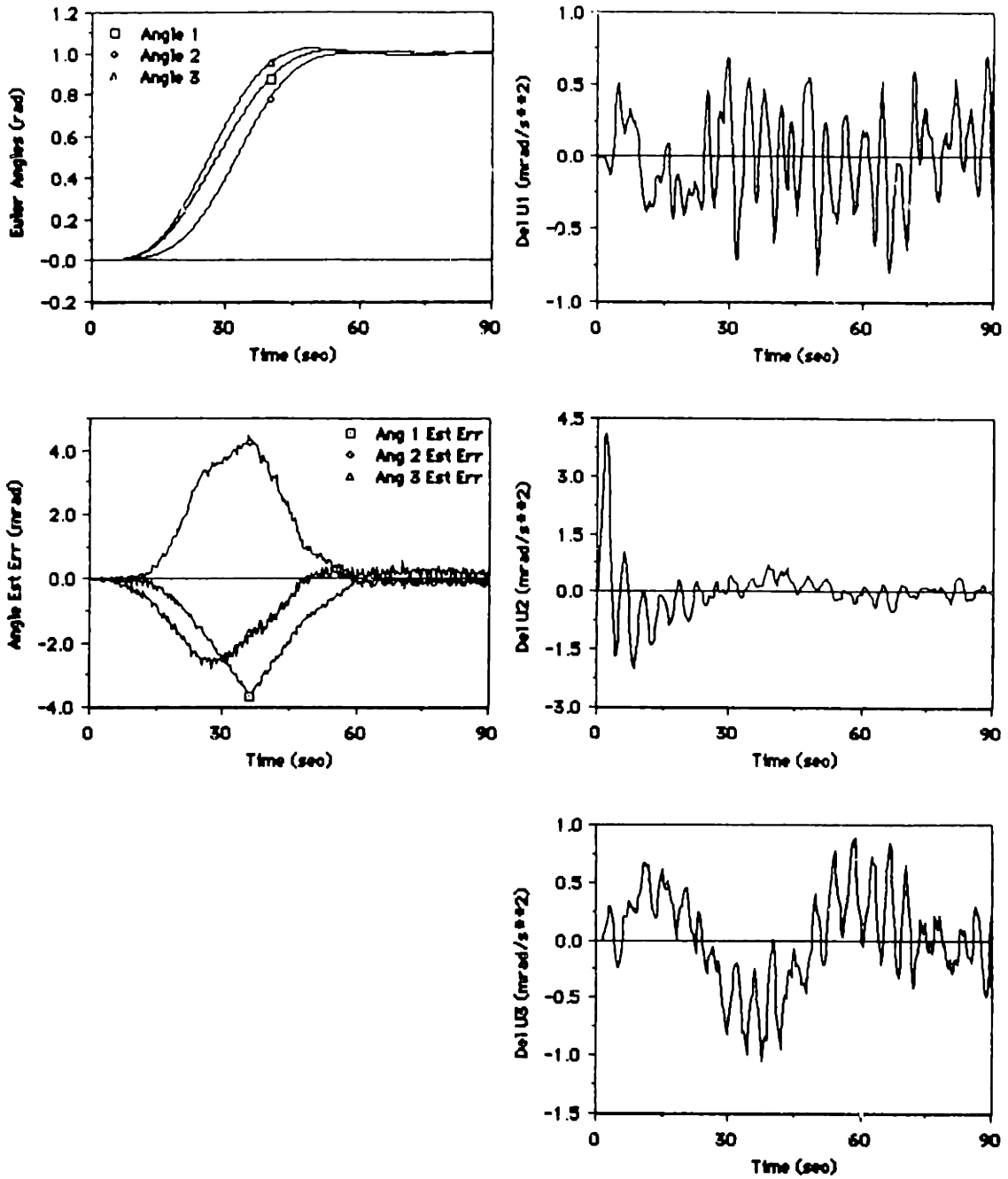


Figure 4-17. Euler Angles, Angle Estimate Errors, and Pseudo-Control Corrections for Case 6

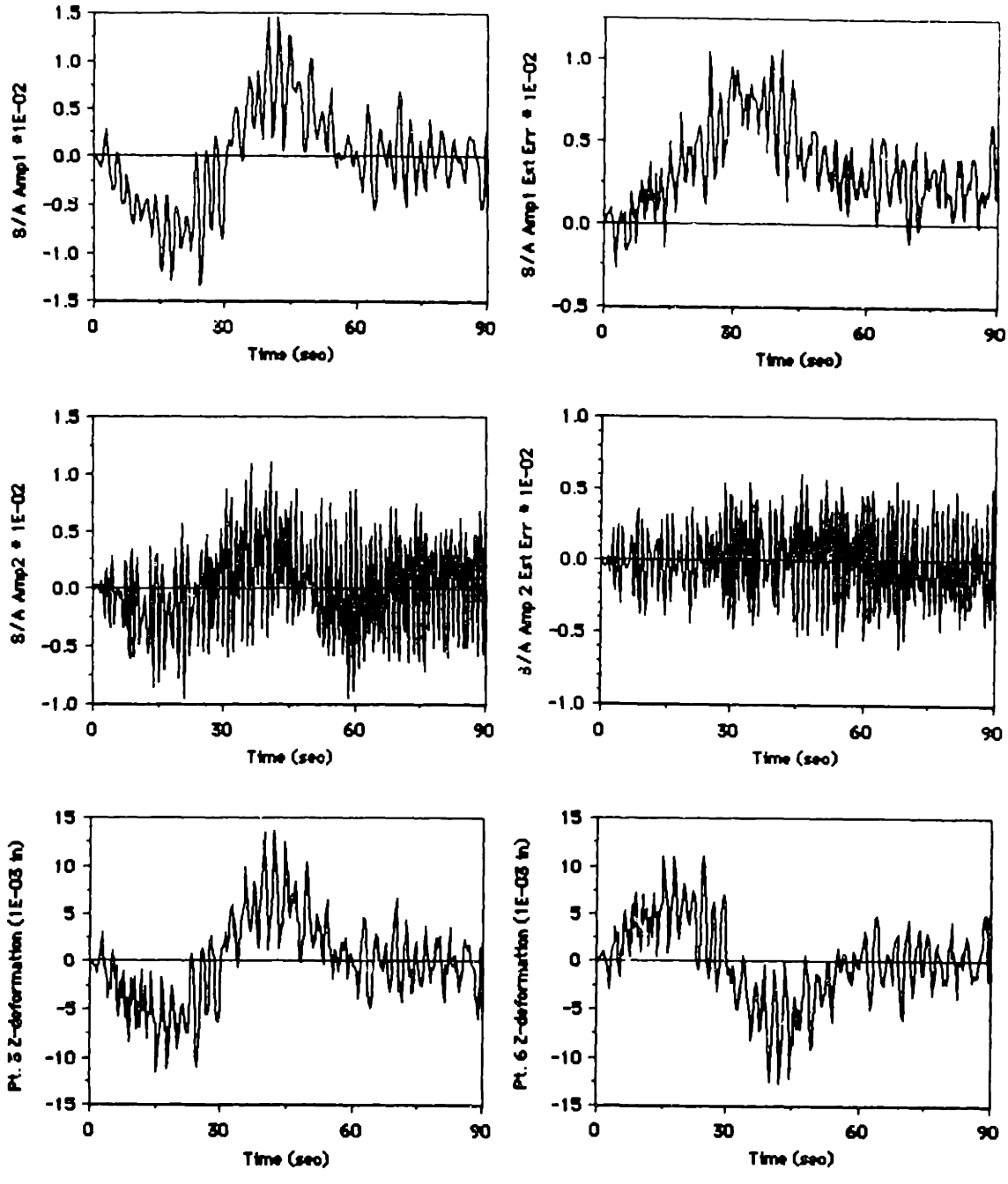


Figure 4-18. Solar Array Modal Amplitudes, Their Estimate Errors, and Sensor Point Deformations for Case 6

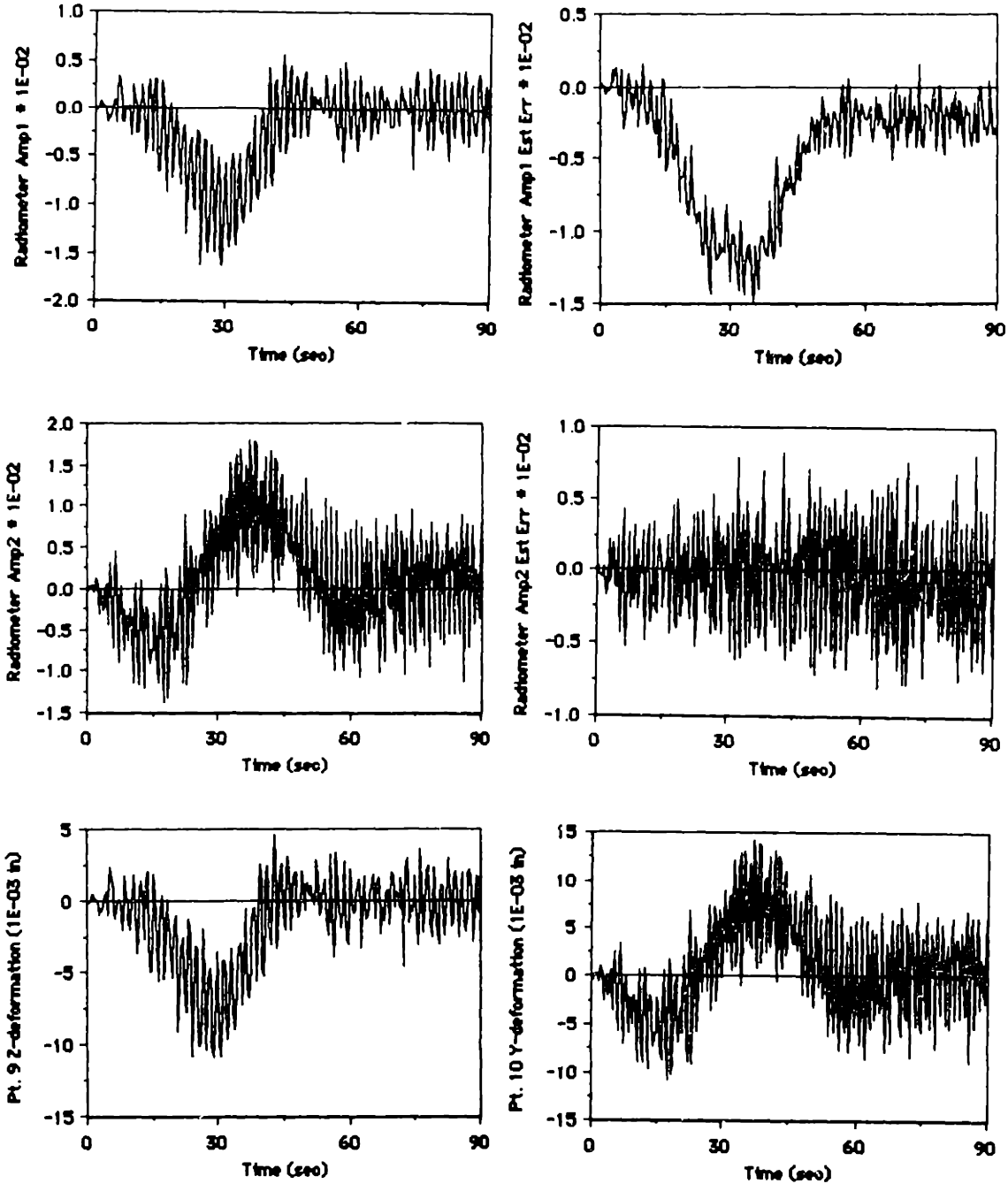


Figure 4-19. Radiometer Modal Amplitudes, Their Estimate Errors, and Sensor Point Deformations for Case 6

and

$$\sigma^2(w_3) = 1.5 \times 10^{-5} ,$$

where w_1, w_2, and w_3 are the disturbances introduced into the angular acceleration. The Euler angle profiles shown in Figure 4-17 appear identical to those of Case 5. The angle estimate errors have profiles similar to Case 5, but with higher frequency content. The pseudo-control corrections bear little resemblance to those in Case 5. Figure 4-18 shows the amplitudes of the controlled solar array modes and their estimate errors. The estimate of the first solar array modal amplitude seems to match the shape of the actual modal amplitude fairly well. The second solar array modal amplitude shows a great deal of high frequency content from the process noise. Figure 4-19 shows the amplitudes of the two controlled radiometer modes and their estimate errors. As in Case 5, the estimate for the first radiometer mode seems to be quite poor. The higher frequency content is evident in all radiometer modal amplitudes.

Case 7 (Figures 4-20 through 4-23) is the same as Case 6, except that the initial angles of the spacecraft are off by 0.05 radians. Figure 4-20 shows the off-nominal initial conditions in the Euler angles. The sensor point deformations are about twice as high as in Case 5, and have a much higher frequency content. Figure 4-21 shows that the peak pseudo-control corrections are nearly the same as those in the equivalent Case 4 of Section 4.4. The pseudo-control profiles are also similar, except that for this case, the undulations do not die out after the end of the maneuver. The angle estimate errors shown in Figure 4-21 indicate that the large initial errors are quickly reduced. Figure 4-22 shows the amplitudes of the two controlled modes of the solar array, and their estimate errors. The results are similar to those shown for Case 2, except that there is a small spike near the initial time, due to the off-nominal initial angles. Figure 4-23 shows the amplitudes of the two controlled modes of the radiometer, and their estimate errors. The first mode estimate seems to be poor, as in the previous cases, while the second mode estimate seems to be fairly good.

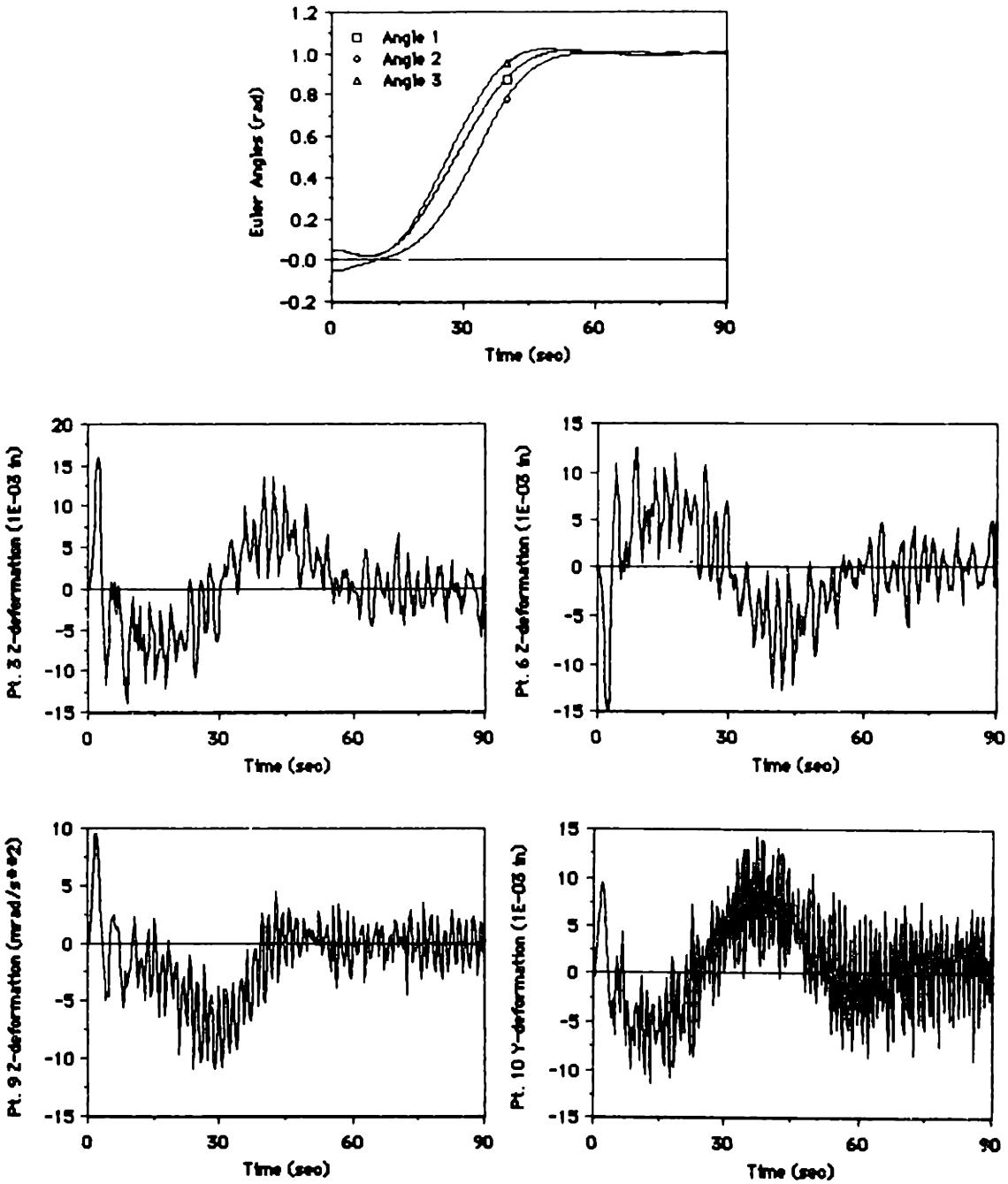


Figure 4-20. Euler Angles and Sensor Point Deformations for Case 7

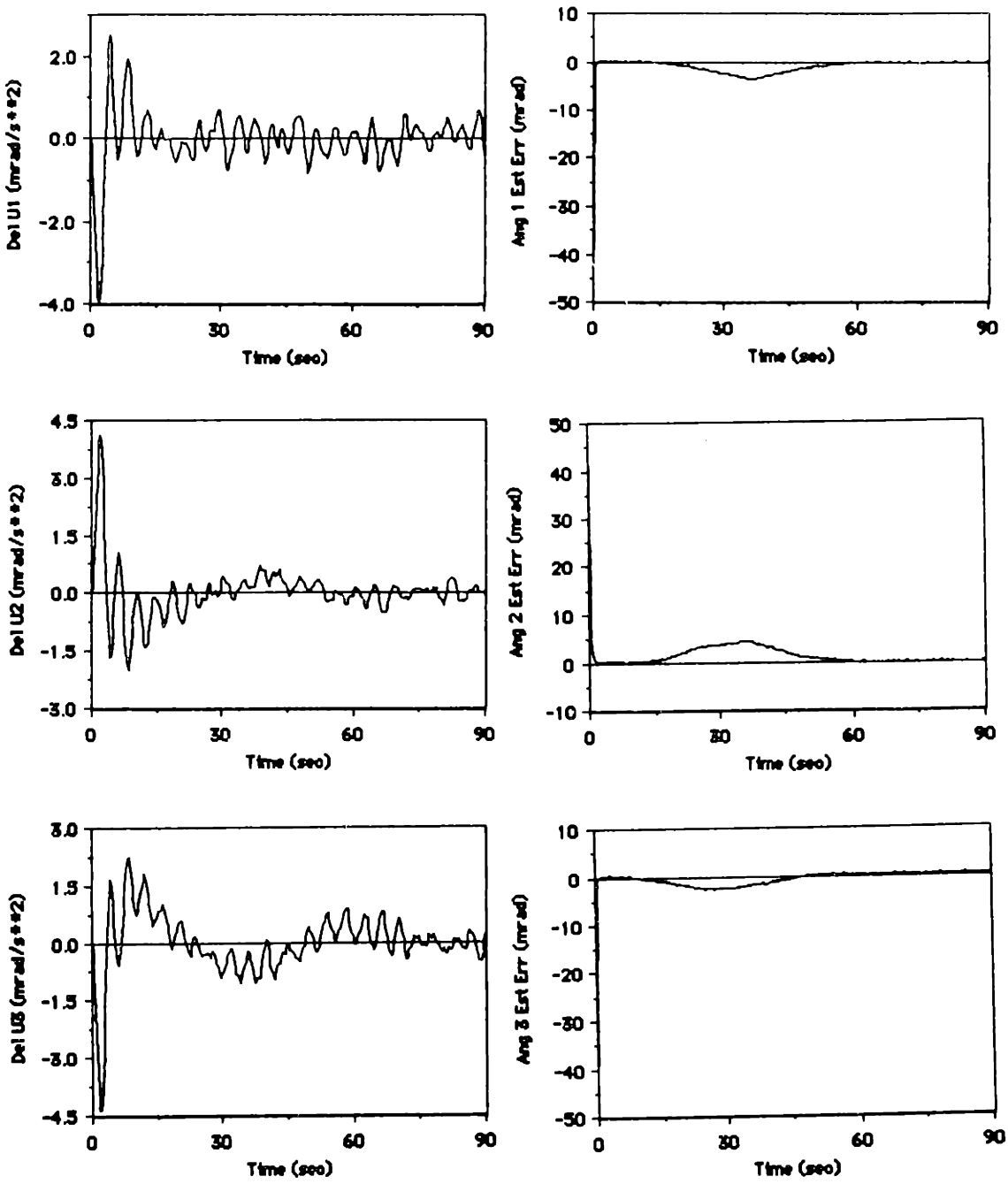


Figure 4-21. Pseudo-Control Corrections and Angle Estimate Errors for Case 7

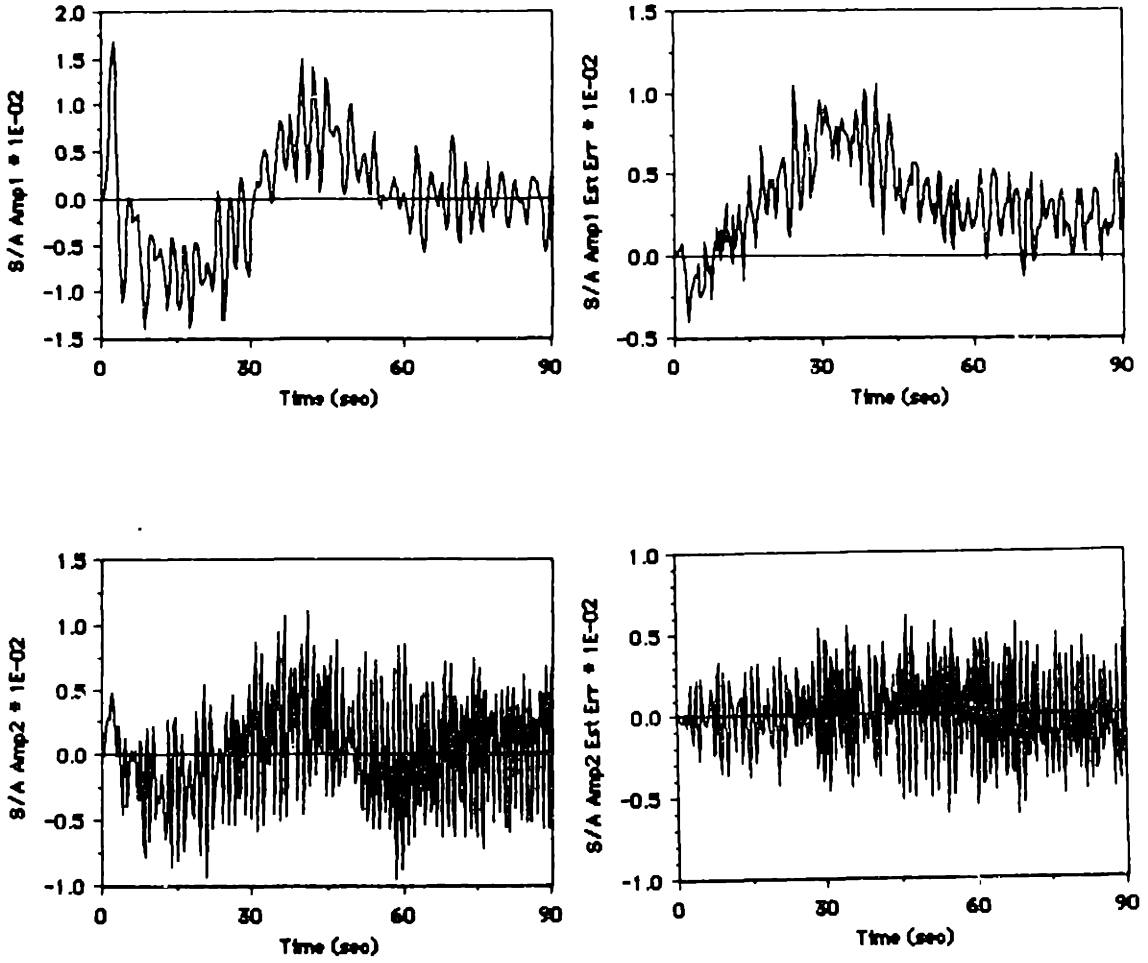


Figure 4-22. Solar Array Modal Amplitudes and Their Estimate Errors for Case 7

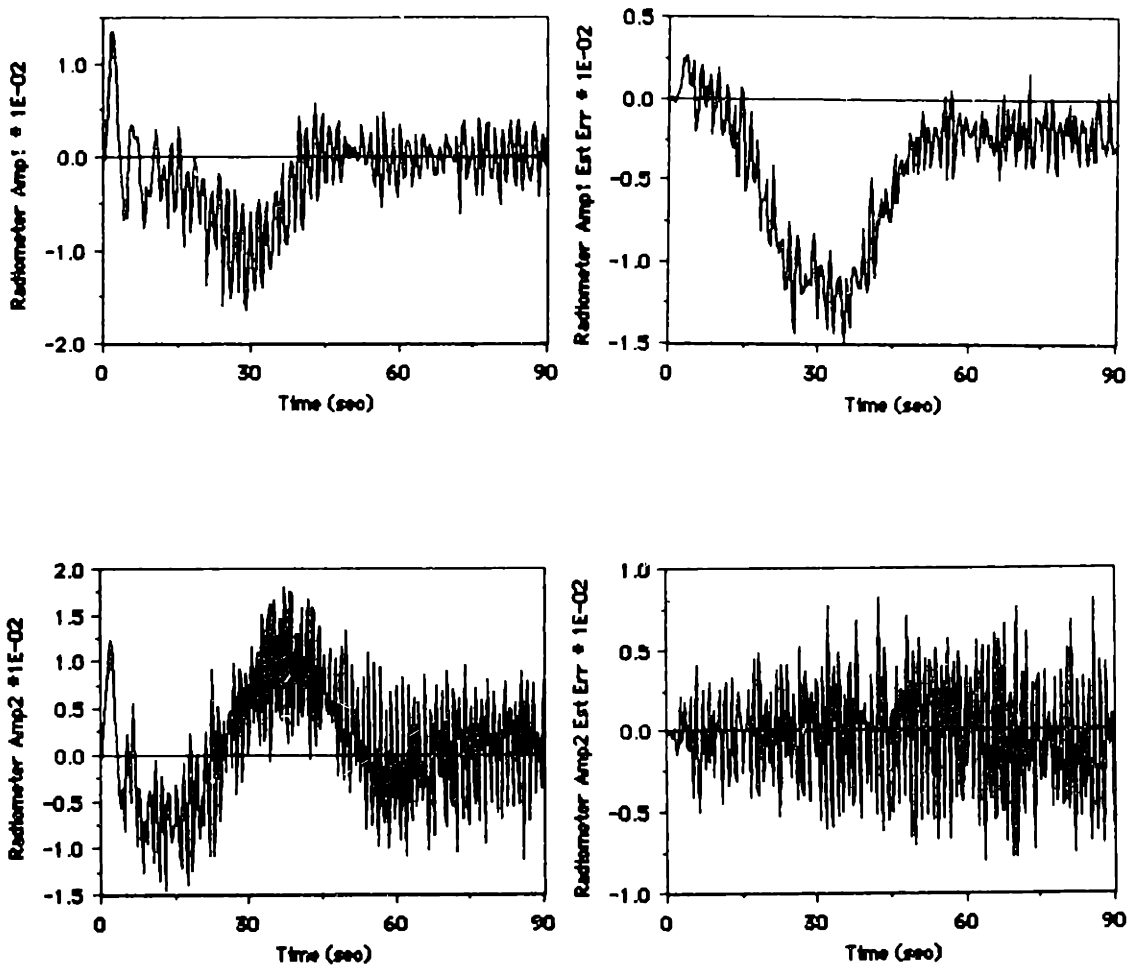


Figure 4-23. Radiometer Modal Amplitudes and Their Estimate Errors for Case 7

4.6 Conclusions

This chapter has presented a formulation for general nonlinear slewing maneuvers for flexible spacecraft, whereby a rigid body nominal control profile is applied while a perturbation feedback controller limits the flexible body response and controls the plant to follow the rigid body nominal trajectory. The use of control smoothing in both the rigid body nominal solution and the perturbation feedback controller greatly reduces the excitation to the elastic degrees of freedom. The numerical results of Section 4.3 show that the break frequency used for the control smoothing formulation should be linked to the maneuver time in order to produce good results. The results of Section 4.4 show that the perturbation feedback controller performs very well under off-nominal conditions for a 60 second maneuver. For further research, it is recommended that the maneuver time be shortened so that the break frequency, ω_B, overlaps some of the structural frequencies. Such a case should prove challenging, since this would involve more interaction between the rigid modes and the elastic modes. The results of Section 4.5 demonstrates the use of a modified Kalman filter, using the same linearized plants as in the perturbation feedback controller. For further research, it is recommended that a sensor location optimization be performed to minimize the possibility of observation spillover. Furthermore, a perturbation estimation approach may be used, whereby one estimates the state perturbations rather than the states themselves.

For maneuvers where the desired final conditions are different from the nominal final conditions, one must use the optimal perturbation feedback of Chapter 3, with the modifications described in Section 3.7 for the nonlinear plant. Such an approach would result in near-optimal time-varying feedback gains, with the same type of linearized plant as used for the results of this chapter.

CHAPTER 5

SUMMARY AND CONCLUSIONS

This thesis has covered three different, yet inter-related topics. Chapter 2 has dealt with a new class of closed-form solutions for finite-time linear-quadratic optimal control problems. These closed-form solutions are used in Chapter 3, which presents the solution for the neighboring extremal path problem, as applied to spacecraft slewing maneuvers. Chapter 4 has dealt with general nonlinear slewing maneuvers for flexible spacecraft, for which the results of Chapter 3 are useful when the terminal conditions are slightly perturbed. A more detailed summary of each chapter follows.

5.1 Summary and Conclusions for Chapter 2

Chapter 2 has dealt with a new class of closed-form solutions for finite-time linear-quadratic optimal control problems where the plant is linear time-invariant. This class of closed-form solutions is based on Potter's solution, which consists of a steady-state plus transient term, for the differential matrix Riccati equation. Five basic differential equations are identified for the solution of finite-time linear-quadratic optimal control problems. Closed-form solutions are presented for these five basic differential equations, and example control problems are presented where these solutions are used to obtain closed-form analytic expressions for the feedback gains, state trajectories, control trajectories, and residual state trajectories, with the assumptions of perfect plant knowledge, and perfect state estimation.

For each example control problem, comparisons are made with closed-form solutions based on the Kalman-Englar method, and on the state transition matrix. For each case, it is found that the new class of closed-form solutions is more efficient than the Kalman Englar type of solution based on the state transition matrix. Furthermore, it is well

known that the Kalman-Englar solution for the Riccati matrix is numerically unstable when the propagation time-step is large, or when the Riccati solution is not symmetrized at each time-step. Such numerical problems do not occur in Potter's solution for the Riccati matrix. Thus, it seems that the new class of solutions is numerically superior to the Kalman-Englar type of solutions for feedback gains and state transition matrix solutions for state and control trajectories. However, a rigorous analysis of the numerical stability and error propagation characteristics of the new class of closed-form solutions remains a topic for further research.

The relationship between the new class of solutions and the state transition matrix solutions is illustrated by means of reducing subspace transformations for the Hamiltonian matrix.

5.2 Summary and Conclusions for Chapter 3

The closed-form solutions developed in Chapter 2 are applied to the free-final-time neighboring extremal path problem with linear terminal constraints, a quadratic performance index, and a linear time-invariant plant. Closed-form solutions are presented for the perturbation feedback gains which cause the system to follow a neighboring extremal path when subjected to small perturbations in the initial conditions and terminal constraints. Numerical experiments indicate that slight numerical modifications can greatly reduce the sensitivity of the feedback gains near the final time. An extension is shown for using the closed-form solutions for problems with nonlinear plants.

5.3 Summary and Conclusions for Chapter 4

Chapter 4 has presented a formulation for general nonlinear slewing maneuvers for flexible spacecraft, whereby a rigid body nominal control profile is applied while a perturbation feedback controller limits the flexible body response and controls the plant to follow the rigid body nominal trajectory. The use of control smoothing in both the rigid body nominal solution and the perturbation feedback controller greatly reduces

the excitation to the elastic degrees of freedom.

Numerical results show that the break frequency used for the control smoothing formulation for the rigid body nominal solution should be linked to the maneuver time in order to produce good results. Numerical results for the perturbation feedback controller show that it performs very well under off-nominal conditions for a 60 second maneuver. For further research, it is recommended that the maneuver time be shortened so that the break frequency for the rigid body nominal solution overlaps some of the structural frequencies. Such a case should prove challenging, since this would involve more interaction between the rigid modes and the elastic modes.

A modified Kalman filter is presented for estimating the system states. Numerical results indicate that the approach is feasible. However, for further work, it is recommended that a sensor location optimization be performed to minimize the possiblility of observation spillover. Furthermore, a perturbation estimation approach may be used, whereby one estimates the state perturbations rather than the states themselves.

For maneuvers where the desired final conditions are different from the nominal final conditions, one must use the optimal perturbation feedback of Chapter 3, with the modifications for nonlinear plants. Such an approach would result in near-optimal time-varying feedback gains, using the same type of linearized plant as used for the results of Chapter 4.

LIST OF REFERENCES

[1] Alfriend, K.T., Longman, R.W., and Bercaw, W.S., "On Frequency Response Interpretations of Optimal Slewing Maneuvers," Proceedings of the Second VPI & SU/AIAA Symposium on Dynamics and Control of Large Flexible Spacecraft, Blacksburg, Virginia, June 1979, pp. 65-79.

[2] Aspinwall, D.W., "Acceleration Profiles for Minimizing Residual Response," Journal of Dynamic Systems, Measurement, and Control, Vol. 102, March 1980.

[3] Athans, M. and Falb, P.L., Optimal Control, McGraw-Hill, New York, 1966.

[4] Aubrun, J., Breakwell, J., Gupta, N., Lyons, M., and Margulies, G., "ACOSS FIVE (Active Control of Space Structures) Phase 1A," RADC-TR-82-21, March 1982.

[5] Bartels, R.H. and Stewart, G.W., "A Solution of the Equation AX + XB = C," Communications of ACM, Vol. 15, No. 9, September 1972, pp. 820-826.

[6] Bauer, T.P., Wood, L.J., and Caughey, T.K., "Gain Indexing Schemes for Low-Thrust Perturbation Guidance," Journal of Guidance, Control, and Dynamics, Vol. 6, No. 6, November-December 1983, pp. 518-525.

[7] Bodden, D.S., and Junkins, J.L., "Eigenvalue Optimization Algorithms for Structure/Controller Design Iterations," Journal of Guidance, Control, and Dynamics, Vol. 8, No. 6, November-December 1985, pp. 697-706.

[8] Bodley, C.S., Devers, A.D., Park A.C., and Frisch, H.P., "A Digital Computer Program for the Dynamic Interaction Simulation of Controls and Structure (DISCOS)," NASA Technical Paper 1219, Vol. 1, 1978.

[9] Breakwell, J.A., "Optimal Control of Distributed Systems," Paper No. 80-1737, presented at the AIAA Guidance and Control Conference, Danvers, Massachusetts, August 1980.

[10] Breakwell, J.A., "Optimal Feedback Slewing of Flexible Spacecraft," Journal of Guidance and Control, Vol. 4, No. 5, September-October 1981, pp. 472-479.

[11] Breakwell, J., Speyer, J.L., and Bryson, A.E., Jr., "Optimization and Control of Nonlinear Systems Using the Second Variation," Journal of the Society for Industrial and Applied Mathematics, Vol. 1, Ser. A, No. 2, 1963, pp. 193-223.

[12] Brockett, R.W., Finite Dimensional Linear Systems, John Wiley and

Sons, Inc., New York, New York, 1970.

[13] Brown, M.E., "Rapid Slewing Maneuvers of a Flexible Spacecraft Using On/Off Thrusters," Master of Science Thesis, Massachusetts Institute of Technology, September 1983.

[14] Bryson, A.E., Jr. and Ho, Y.C., Applied Optimal Control, Hemisphere Publishing Corp., Washington, D.C., 1975.

[15] Carrington, C.K., and Junkins, J.L., "Nonlinear Feedback Control of Spacecraft Slew Maneuvers," The Journal of the Astronautical Sciences, Vol. 32, No. 1, January-March 1984, pp. 29-45.

[16] Carrington, C.K., and Junkins, J.L., "Optimal Nonlinear Feedback Control of Spacecraft Attitude Maneuvers," Journal of Guidance, Control, and Dynamics, Vol. 9, No. 1, pp. 99-107, January-February 1986.

[17] Chen, J., and Kane T.R., "Slewing Maneuvers of Gyrostat Spacecraft," The Journal of the Astronautical Sciences, Vol. 28, No. 3, July-September 1980, pp. 267-281.

[18] Chun, H.M., "Optimal Distrubuted Control of a Flexible Spacecraft During a Large-Angle Maneuver," Master of Science Thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1982.

[19] Chun, H.M. and Turner, J.D., "A Simple Algorithm for the Selection of Terminal Penalty Weight Matrices," Journal of Guidance, Control, and Dynamics, Vol. 9, No. 4, July-August 1986, pp. 503-505.

[20] Chun, H.M., Turner, J.D., and Juang, J.-N., "Disturbance-Accommodating Tracking Maneuvers of Flexible Spacecraft," The Journal of the Astronautical Sciences, Vol. 33, No. 2, April-June, 1985, pp. 197-216.

[21] Chun, H.M., Turner, J.D., and Juang, J.-N., "Spacecraft Slewing Maneuvers Using a Closed-Form Solution for the Neighboring Extremal Path Problem," AIAA/AAS Astrodynamics Conference, Vail, Colarado, August 12-15, 1985.

[22] Chun, H.M., Turner, J.D., and Juang, J.-N., "Frequency-Shaped Large-Angle Maneuvers for Flexible Spacecraft," (Progress Report), AIAA/AAS Astrodynamics Conference, Vail, Colarado, August 12-15, 1985.

[23] Cochran, J.E., and Junkins, J.L., "Large Angle Satellite Attitude Maneuvers," Proceedings of the Flight Mechanics/Estimation Theory Symposium, Goddard Space Flight Center, Greenbelt, Maryland, April, 1975.

[24] Davison, E.J., "The Numerical Solution of $\dot{X} = A_1X + XA_2 + D$, $X(0) = C$," IEEE Transactions on Automatic Control, Vol. AC-20, No. 4, August 1975, pp. 566-567.

[25] Dennis, J.E., Jr., and Schnabel, R.B., "Least Change Secant Updates for Quasi-Newton Methods," SIAM Review, Vol. 21, No. 4, October 1979, pp. 443-459.

[26] Deuflhard, P., Pesch, H.J., and Rentrap, P., "A Modified Continuation Method for the Numerical Solution of Nonlinear Two-Point Boundary-Value Problems by Shooting Techniques," Numer. Math., Vol. 26, 1976, pp. 327-343.

[27] Dwyer, T.A.W. III, "Exact Nonlinear Control of Spacecraft Slewing Maneuvers with Internal Momentum Transfer," Journal of Guidance, Control, and Dynamics, Vol. 9, No. 2, March-April 1986, pp. 240-247.

[28] Dwyer, T.A.W., III, and Batten, A.L., "Exact Spacecraft Detumbling and Reorientation Maneuvers with Gimbaled Thrusters and Reaction Wheels," The Journal of the Astronautical Sciences, Vol. 33, No. 2, April-June 1985, pp. 217-232.

[29] Farrenkopf, R.L., "Optimal Open Loop Maneuvering Profiles for Flexible Spacecraft," Paper No. 78-1280, Proceedings of AIAA Guidance and Control Conference, Palo Alto, California, August 7-9, 1978.

[30] Floyd, M.A., Brown, M.E., Turner, J.D., and Vander Velde, W.E., "Implementation of a Minimum Time and Fuel On/Off Thruster Control System for Flexible Spacecraft," The Journal of the Astronautical Sciences, (to appear).

[31] Frisch, H.P., "A Vector-Dyadic Development of the Equations of Motion for N-Coupled Flexible Bodies and Point Masses," NASA Technical Note TN D-8047, August 1975.

[32] Goldfarb, D., "A Family of Variable-Metric Methods Derived by Variational Means, Mathematics of Computation, Vol. 24, No. 109, January 1970, pp. 23-26.

[33] Golub, G.H., Nash, S., and Van Loan, C.F., "A Hessenburg-Schur Method for the Problem AX + XB = C," IEEE Transactions on Automatic Control, Vol. AC-24, No. 6, December 1979, pp. 909-913.

[34] Golub, G.H., and Van Loan, C.F., Matrix Computations, The Johns Hopkins University Press, Baltimore, Maryland, 1983.

[35] Greenstadt, J., "Variations on Variable-Metric Methods," Mathematics of Computation, Vol. 24, No. 109, January 1970, pp. 1-22.

[36] Gupta, N.K., "Frequency-Shaped Cost Functionals: Extension of Linear-Quadratic-Gaussian Design Methods," Journal of Guidance and Control, Vol. 3, No. 6, November-December 1980, pp. 529-535.

[37] Gupta, N.K., "Robust Control/Estimator Design by Frequency Shaped

Cost Functionals," presented at the 20th Conference on Design and Control, San Diego, California, 1981.

[38] Hefner, R., Kawauchi, B., Melzer, S., and Williamson, R., "A Terminal Controller for the Pointing of Flexible Spacecraft," Paper No. 80-1670, presented at the AIAA Astrodynamics Conference, Danvers, Massachusetts, August 11-13, 1980.

[39] Hegg, D.R., et al, "ACOSS Eleven-Fourth Semiannual Technical Report," Vol. 2, CSDL-R-1648, Contract No. F30602-81-C-0180, The Charles Stark Draper Laboratory, Inc., Section 2, August 1983.

[40] Ho, J.Y.L., "Direct Path Method for Flexible Multibody Spacecraft Dynamics," Journal of Spacecraft and Rockets, Vol. 14, February 1977, pp. 102-110.

[41] Ho, J.Y.L., and Herber, D.R., "Development of Dynamics and Control Simulation of Large Flexible Space Systems," Journal of Guidance, Control, and Dynamics, Vol. 8, No. 3, May-June 1985, pp. 374-383.

[42] Hughes, P.C., and Fung, J.C., "Liapunov Stability of Spinning Satellites with Long Flexible Appendages," Journal of Celestial Mechanics, Vol. 4, 1971, pp. 295-308.

[43] Hurty, W.C., and Rubinstein, M.F., Dynamics of Structures, Prentice-Hall, 1964.

[44] Itô, K., "On Stochastic Differential Equations," Memoirs of the American Mathematical Society, Vol. 4, 1951.

[45] Jamshidi, A.H., "An Overview of the Solutions of the Algebraic Matrix Riccati Equation and Related Problems," Large Scale Systems: Theory and Application, North-Holland Publishing Co., Vol. 1, No. 3, August 1980, Singh M.G. and Sage, A.P. (editors), pp. 167-192.

[46] Jazwinski, M., Stochastic Processes and Filtering Theory, Academic Press, New York, New York, 1970.

[47] Juang, J.-N., Turner, J.D., and Chun, H.M., "Closed-Form Solutions of Control Gains for a Terminal Controller," VPI&SU/AIAA 4th Symposium on Dynamics and Control of Large Structures, June 6-8, 1983, Blacksburg, Virginia.

[48] Juang, J.-N., Turner, J.D., and Chun, H.M., "Closed-Form Solutions for Feedback Control with Terminal Constraints," Journal of Guidance, Control, and Dynamics, Vol. 8, No. 1, January-February 1985, pp. 39-43.

[49] Juang, J.-N., Turner, J.D., and Chun, H.M., "Closed-Form Solutions for a Class of Optimal Quadratic Regulator Problems with Terminal Constraints," Journal of Dynamic Systems, Measurement, and Control, Vol. 108, No. 1, March 1986, pp. 44-48.

[50] Junkins, J.L., "Comment on Optimal Feedback Slewing of Flexible

Spacecraft," Journal of Guidance, Control, and Dynamics, Vol. 5, No. 3, May-June 1982, p. 318.

[51] Junkins J.L., Bodden, D.S., and Turner, J.D., "A Unified Approach to Structures and Control System Design Iterations," presented to the Fourth International Conference on Applied Numerical Modeling, Tainan, Taiwan, December 27-29, 1984.

[52] Junkins, J.L., and Turner, J.D., "Optimal Continuous Torque Attitude Maneuvers," Journal of Guidanc and Control, Vol. 3, No. 3, May-June 1980, pp. 210-217.

[53] Kalman, R.E., and Bucy, R.S., "New Results in Linear Filtering and Prediction Theory," Journal of Basic Engineering, Transactions of ASME, Ser. D, Vol. 83, 1961, pp. 95-108.

[54] Kalman, R.E., and Englar, T.S., "A User's Manual for the Automatic Synthesis Program," NASA Report CR-475.

[55] Kane, T.R., Ryan, R.R., and Banerjee, A.K., "Dynamics of a Beam Attached to a Moving Base," Paper AAS 85-390, AAS/AIAA Astrodynamics Specialist Conference, Vail, Colorado, August 12-15, 1985.

[56] Kaza, K.R.V., and Kvaternik, R.G., "Nonlinear Flap-Lag-Axial Equations of a Rotating Beam," AIAA Journal, Vol. 15, No. 6, June 1977, pp. 871-874.

[57] Kelley, H.J., "An Optimal Guidance Approximation Theory," IEEE Transactions on Automatic Control, Vol. AC-9, October 1964, pp. 375-380.

[58] Kwakernaak, H., and Sivan, R., Linear Optimal Control Systems, John Wiley and Sons, Inc., New York, New York, 1972.

[59] Lassen, H.A., and Elliot, L.E., "Use of Control of Higher Order Derivatives of Excitation to Minimize Dynamic Response," TRW IOC, January 7, 1977.

[60] Longman, R.W., and Alfriend, K.T., "Optimal Control of Large Angle Maneuvers for Flexible Spacecraft," Proceedings of the 8th IFAC Symposium, Oxford, England, July 1979, pp. 57-62.

[61] Markley, F.L., "Large Angle Maneuver Strategy for Flexible Spacecraft," Paper No. 79-156, AAS/AIAA Astrodynamics Specialist Conference, Provincetown, Massachusetts, June 25-27, 1979.

[62] Martensson, K., "On the Matrix Riccati Equation," Information Sciences, Vol. 3, 1971, pp. 17-49.

[63] McReynolds, S.R., and Bryson, A.E., Jr., "Successive Sweep Method for Solving Optimal Programming Problems," Sixth Joint Automatic Control Conference, Troy, New York, June 1965.

[64] Meirovitch, L., Analytical Methods in Vibrations, MacMilan Co., New York, New York, 1967.

[65] Moler, C., and Van Loan, C.F., "Nineteen Dubious Ways to Compute the Exponential of a Matrix," SIAM Review, Vol. 20, No. 4, October 1978, pp. 801-836.

[66] Moore, J.B., and Anderson, B.D.O., "Optimal Linear Control Systems with Input Derivative Constraints," Proceedings of the IEEE, Vol. 114, No. 12, December 1967, pp. 1987-1990.

[67] Osborne, M.R., "On Shooting methods for Boundary Value Problems," Journal of Mathematical Analysis and Applications, Vol. 27, 1969, pp. 417-433.

[68] Öz, H., and Özgüner, Ü., "Variable Structure System Control of Flexible Spacecraft," AIAA Paper 84-2002, August 1984.

[69] Pace, I.S., and Barnett, S., "Comparison of Numerical Methods for Solving Lyapunov Matrix Equations," International Journal of Control, Vol. 15, No. 5, 1972, pp. 907-915.

[70] Pesch, H.J., "Numerical Computation of Neighboring Optimum Feedback Control Schemes in Real-Time," Applied Mathematics and Optimization, Vol. 5, 1979, pp. 231-252.

[71] Potter, J.E., "Matrix Quadratic Solutions," SIAM Journal of Applied Mathematics, Vol. 14, No. 3, 1964, pp. 496-501.

[72] Potter, J.E., "A Matrix Equation Arising in Statistical Filter Theory," NASA CR-270, 1965.

[73] Potter, J.E., and Vander Velde, W.E., "Optimal Mixing of Gyroscope and Star Tracker Data," Journal of Spacecraft and Rockets, Vol. 5, May 1968, pp. 536-540.

[74] Powers, W.F., "Techniques for Improved Convergence in Neighboring Optimum Guidance," AIAA Journal, Vol. 8, No. 12, December 1970, pp. 2235-2241.

[75] Prussing, J.E., "A Simplified Method for Solving the Matrix Riccati Equation," International Journal of Control, Vol. 15, No. 5, 1972, pp. 995-1000.

[76] Richter, S.L., and DeCarlo, R.A., "Continuation Methods: Theory and Applications," IEEE Transactions on Circuits and Systems, Vol. CAS-30, No. 6, June 1983, pp. 347-352.

[77] Roberts, S.M., Shipman, J.S., and Roth, C.V., "Continuation in Quasilinearization," Journal of Optimization Theory and Applications, Vol. 2, No. 3, 1968, pp. 164-178.

[78] Roberts, S.M., and Shipman, J.S., Two-Point Boundary Value Problems: Shooting Methods, American Elsevier Publishing Company,

New York, New York, 1972.

[79] Schmidt, W.F., "Adaptive Step Size Selection for Use with the Continuation Method," International Journal for Numerical Methods in Engineering, Vol. 12, 1978, pp. 677-694.

[80] Serbin, S.M., and Serbin, C.A., "A Time-Stepping Procedure for $X = A_1X + XA_2 + D$, $X(0) = C$," IEEE Transactions on Automatic Control, Vol AC-25, No. 6, December 1980, pp. 1138-1141.

[81] Singh, VanderVoort, R.J., and Likins, P.W., "Dynamics of Flexible Bodies in Tree Topology -- A Computer Oriented Approach," Paper No. AIAA-84-1024, AIAA/ASME/ASCE 25th Structures, Structural Dynamics and Materials Conference, Palm Springs, California, May 14-18, 1984.

[82] Skaar, S.B., and Kraige, L.G., "Large-Angle Spacecraft Attitude Maneuvers Using an Optimal Reaction Wheel Power Criterion," The Journal of the Astronautical Sciences, Vol. 32, No. 1, January-March 1984, pp. 47-61.

[83] Skaar, S.B., and Tucker, D., "The Optimal Control of Flexible Systems Using a Convolution Integral Description of Motion," Proceedings of the 22nd IEEE Conference on Decision and Control, San Antonio, Texas, December 14-16, 1983.

[84] Skelton, R., and Likins, P.W., "Orthogonal Filters for Model Error Compensation in the Control of Nonrigid Spacecraft," Journal of Guidance and Control, Vol. 1, No. 1, January 1978, pp. 41-49.

[85] Speyer, J.L., and Bryson, A.E., Jr., "A Neighboring Optimum Feedback Control Scheme Based on Estimated Time-to-Go with Application to Re-Entry Flight Paths," AIAA Journal, Vol. 6, No. 5, May 1968, pp. 769-776.

[86] Swigert, C.J., "Shaped Torques Technique," Paper No. 78-1692, AIAA Conference on Large Space Platforms: Future Needs and Capabilities, Los Angeles, California, September 27-29, 1978.

[87] Thompson, R.C., and Junkins, J.L., "Vibration Control of Flexible Bodies with Translation/Rotational Coupling," presented at the 23rd AIAA Aerospace Sciences Meeting, Reno, Nevada, January 1985.

[88] Thompson, R.C., and Junkins, J.L., "A Quasi-Analytical Perturbation Method for Open Loop Control of Nonlinear Systems," presented to the Mountain Lake and Control Institute, Dynamics and Control, ed. by J.L. Junkins, pp. 16-36, published by VPI&SU, Blacksburg, Virginia, June 1985.

[89] Turner, J.D., "Optimal Large Angle Maneuvers for Large Flexible Space Structures," Ph.D. Dissertation, VPI&SU, Blacksburg, Virginia, 1980.

[90] Turner, J.D., Chun, H.M., et al., "ACOSS Eleven Semiannual

Technical Report," CSDL-R-1536, Contract No. F30602-81-C-0180, The Charles Stark Draper Laboratory, Inc., February 1982.

[91] Turner, J.D., and Chun, H.M., "Optimal Distributed Control of a Flexible Spacecraft Using Control-Rate Penalties in the Controller Design," Paper No. 82-1438, presented at the AIAA Astrodynamics Conference, San Diego, California, August 9-11, 1982.

[92] Turner, J.D., and Chun, H.M., "Optimal Feedback Control of a Flexible Spacecraft During a Large-Angle Rotational Maneuver," Paper No. 82-1589, presented at the AIAA Guidance and Control Conference, San Diego, California, August 9-11, 1982.

[93] Turner, J.D., and Chun, H.M., "Optimal Distributed Control of a Flexible Spacecraft During a Large-Angle Rotational Maneuver," Journal of Guidance, Control, and Dynamics, Vol. 7, No. 3, May-June 1984, pp. 257-264.

[94] Turner, J.D., Chun, H.M., and Juang, J.-N., "An Analytic Solution for the State Trajectories of Feedback Control Systems," Journal of Guidance, Control, and Dynamics, Vol. 8, No. 1, January-February 1985, pp. 147-148.

[95] Turner, J.D., Chun, H.M., and Juang, J.-N., "Closed-Form Solutions for a Class of Optimal Quadratic Tracking Problems," Journal of Optimization Theory and Applications, Vol. 47, No. 4, December 1985, pp. 465-481.

[96] Turner, J.D., Chun, H.M., Juang, J.-N., and Junkins, J.L., "Computational Methods for Closed-Loop Control Problems," presented at the 23rd AIAA Aerospace Sciences Conference, January 1986.

[97] Turner, J.D., Chun, H.M., and Junkins, J.L., "Optimal Large-Angle Maneuvers with Vibration Supression," Paper presented at the Workshop on Modelling, Analysis and Optimization Issues for Large Space Structures, Williamsburg, Virginia, May 13-14, 1982.

[98] Turner, J.D., and Junkins, J.L., "Optimal Large Angle Maneuvers with Simultaneous Shape Control/Vibration Arrest," Proceedings of the Fourth Annual Flight Mechanics/Estimation Theory Symposium, NASA, GSFC, October 17-18, 1979, Greenbelt, Maryland.

[99] Turner, J.D., and Junkins, J.L., "Optimal Large Angle Single-Axis Rotational Maneuvers of Flexible Spacecraft," Journal of Guidance and Control, Vol. 3, No. 6, November-December 1980, pp. 578-585.

[100] Turner, J.D., Messac, A., and Junkins, J.L., "Finite Time Matrix Convolution Integral Sensitivity Calculations," Journal of Guidance, Control, and Dynamics, (to appear).

[101] Turner, J.D., Mozzicato, R., and Chun, H.M., "An Analytic Solution for the State when the Plant Input is Specified as a Fourier Series," Journal of Dynamic Systems, Measurement and Control, (to appear).

[102] Vadali, S.R., "Solution of Two-Point Boundary-Value Problems of Optimal Spacecraft Maneuvers," Ph.D. dissertation, Virginia Polytechnical Institute, Blacksburg, Virginia, 1983.

[103] Vadali, S.R., "Variable-Structure Control of Spacecraft Large-Angle Maneuvers," Journal of Guidance, Control and Dynamics, Vol. 9, No. 2, March-April 1986, pp. 235-239.

[104] Vadali, S.R., and Junkins, J.L., "Optimal Open-Loop and Stable Feedback Control of Rigid Spacecraft Attitude Maneuvers," The Journal of the Astronautical Sciences, Vol. 31, No. 2, April-June 1984, pp. 105-112.

[105] Vadali, S.R., and Junkins, J.L., "Spacecraft Large Angle Rotational Maneuvers with Optimal Momentum Transfer," The Journal of the Astronautical Sciences, Vol. 32, No. 2, April-June, 1983, pp. 217-235.

[106] Vadali, S.R., Kraige, L.G., and Junkins, J.L, "New Results on the Optimal Spacecraft Attitutde Maneuver Problem," Journal of Guidance, Control, and Dynamics, Vol. 7, No. 3, May-June 1984, pp. 378-380.

[107] Vander Velde, W.E., and Carignan, C.R., "Number and Placement of Control System Components Considering Possible Failures," Journal of Guidance, Control, and Dynamics, Vol. 7, No. 6, November-December 1984, pp. 703-709.

[108] Van Loan, C.F., "Computing Integrals Involving Matrix Exponentials," IEEE Transactions on Automatic Control, Vol. AC-23, No. 3, June 1978, pp. 395-404.

[109] Vaughan, D.R., "A Negative Exponential Solution for the Matrix Riccati Equation," IEEE Transactions on Automatic Control, Vol. AC-14, No. 1, 1969, pp 72-75.

[110] Vigneron, F.R., 'Comment on "Mathematical Modeling of Spinning Elastic Bodies for Modal Analysis,"' AIAA Journal, Vol. 13, No. 1, January 1975, pp. 126-127.

[111] Wacker, H., Continuation Methods, Academic Press, New York, New York, 1978.

[112] Ward, R.C., "Numerical Computation of the Matrix Exponential with Accuracy Estimate," SIAM Journal of Numerical Analysis, Vol. 14, No. 4, September 1977.

[113] Wie, B., and Barba, P.M., "Quaternion Feedback for Spacecraft Large Angle Maneuvers," Journal of Guidance, Control, and Dynamics, Vol. 8, No. 3, May-June 1985, pp. 360-365.

APPENDIX A

OPERATION COUNT FOR PROPAGATING THE RICCATI MATRIX

This appendix compares the number of operations required for propagating the $(n \times n)$ Riccati matrix via Potter's method [12,45,62,72,73,75] and via the Kalman-Englar method [54].

For Potter's method, the Riccati matrix is given by

$$P(t) = P_{ss} + Z^{-1}(t) , \tag{A1}$$

where Z(t) is propagated by

$$Z(t+\Delta t) = C + e^{\bar{A}\Delta t} Z(t) e^{\bar{A}^T \Delta t} , \tag{A2}$$

and C is defined as

$$C = Z_{ss} - e^{\bar{A}\Delta t} Z_{ss} e^{\bar{A}^T \Delta t} . \tag{A3}$$

The matrices P_{ss}, Z_{ss}, C, and the matrix exponentials are constant. In performing the triple product of (A2), the first product (whether $e^{\bar{A}\Delta t}Z(t)$ or $Z(t)e^{\bar{A}^T\Delta t}$) requires n^3 flops. The second product requires only $n^3/2$ flops because the result is symmetric, and only about half the matrix need be computed.

The matrix inversion of (A1) is done by a Cholesky decomposition followed by forward elimination and back substitution involving the identity matrix. The Cholesky decomposition is written in the form

$$Z(t) = LL^T , \qquad (A4)$$

where L is lower triangular, and the computation requires about $n^3/6$ flops for a symmetric definite Z(t) [34]. In terms of the Cholesky decomposition, the matrix inversion can be written as

$$LL^TX = I , \qquad (A5)$$

where X represents the inverse of Z(t) and I is the (n × n) identity matrix.

The forward elimination step can be written as

$$LY = I , \qquad (\text{lower} \times \text{lower} = \text{lower}) \qquad (A6)$$

where Y is lower triangular because of the diagonal identity matrix. (The matrix Y is lower triangular for any lower triangular right hand side). Since Y is lower triangular, forward elimination for the i-th column of Y requires $(n-i+1)^2/2$ flops, compared with $n^2/2$ flops for each column of a fully populated Y. Thus, the total amount of work for the forward elimination for Y is $n^3/6$ flops, compared with $n^3/2$ flops for a fully populated right hand side.

Back substitution for X is written as

$$L^TX = Y . \qquad (\text{upper} \times \text{full} = \text{lower}) \qquad (A7)$$

Since X is symmetric, only the lower triangular part need be computed. Back substitution for the appropriate lower partition of the i-th column of X requires $(n-i+1)^2/2$ flops. Thus, back substitution for X requires about $n^3/6$ flops.

The amount of work required for matrix inversion -- decomposition, forward elimination, and back substitution -- is $n^3/2$. Thus, the total

amount of work needed for updating the Riccati matrix -- Z(t) propagation and inversion -- is $2n^3$ for Potter's solution.

For the Kalman-Englar method, the Riccati matrix is propagated by

$$P(t+\Delta t) = [\theta_{21} + \theta_{22}P(t)][\theta_{11} + \theta_{12}P(t)]^{-1} , \tag{A8}$$

where θ_{ij} are constant (n × n) matrices. Computation for the first term in brackets,

$$X_1 = [\theta_{21} + \theta_{22}P(t)] , \tag{A9}$$

requires n^3 flops. Computation for the second term in brackets,

$$X_2 = [\theta_{11} + \theta_{12}P(t)] , \tag{A10}$$

also requires n^3 flops.

Since X_2 is not symmetric, its L-U decomposition,

$$X_2 = LU , \qquad (\square = \triangle\triangledown) \tag{A11}$$

is computed for the solution of (A8). The L-U decomposition with partial pivoting requires $n^3/3$ flops [34]. In terms of the L-U decomposition, the linear equation for $P(t+\Delta t)$ is written as

$$P(t+\Delta t)LU = X_1 . \tag{A12}$$

The above equation may be solved by row-wise forward elimination and back substitution.

The forward elimination step is written as

$$YU = X_1 . \qquad (\square\nabla = \square) \tag{A13}$$

Since X_1 is fully populated, forward elimination for each row of Y requires $n^2/2$ flops, resulting in $n^3/2$ flops for the entire matrix.

Back substitution for $P(t+\Delta t)$ is written as

$$P(t+\Delta t)L = Y . \qquad (\square\triangle = \square) \tag{A14}$$

Since the solution is symmetric, only the upper triangular partition of $P(t+\Delta t)$ need be computed. Back substitution for the appropriate partition of the i-th row of $P(t+\Delta t)$ requires $(n-i+1)^2/2$ flops. Thus, back substitution for the upper triangular part of $P(t+\Delta t)$ requires $n^3/6$ flops.

The linear equation solution -- decomposition, forward elimination, and back substitution -- requires n^3 flops. Thus, the total amount of work required for updating the Riccati matrix -- X_1, X_2 calculation and linear equation solution -- requires $3n^3$ flops for the Kalman-Englar method.

The above operation counts are based on the assumption that only a triangular part of the Riccati matrix is computed. However, for the Kalman-Englar method, numerical problems occur unless the entire matrix is computed and then symmetrized. This is not found to be needed for Potter's solution, perhaps because the inversion of a symmetric definite matrix is numerically more stable than the inversion of a nonsymmetric matrix.

APPENDIX B

INTEGRALS OF EXPONENTIAL MATRICES

Integrals of exponential matrices of the form

$$G(A,B,C,t_2,t_1) = \int_{t_1}^{t_2} e^{A(t_2-\tau)} B e^{C(\tau-t_1)} d\tau \qquad (B1)$$

and

$$H(A,B,C,D,E,t_2,t_1) = \int_{t_1}^{t_2} e^{A(t_2-\tau)} B \int_{t_1}^{\tau} e^{C(\tau-\varepsilon)} D e^{E(\varepsilon-t_1)} d\varepsilon \, d\tau \qquad (B2)$$

are easily computed by using Van Loan's method of exponentiating a higher-dimensioned matrix involving the matrices A, B, C, D, and E (Ref [108]).

To compute the matrix integrals $G(A,B,C,t_2,t_1)$ and $H(A,B,C,D,E,t_2,t_1)$ using Van Loan's method, one forms the matrix

$$\Pi = \begin{bmatrix} A & B & 0 \\ 0 & C & D \\ 0 & 0 & E \end{bmatrix}, \qquad (B3)$$

and exponentiates the matrix scalar product $\Pi(t_2-t_1)$ to yield a matrix of the form

$$e^{\Pi(t_2-t_1)} = \begin{bmatrix} F_1(t_2-t_1) & G_1(t_2-t_1) & H_1(t_2-t_1) \\ 0 & F_2(t_2-t_1) & G_2(t_2-t_1) \\ 0 & 0 & F_3(t_2-t_1) \end{bmatrix}. \qquad (B4)$$

Methods for computing matrix exponentials are discussed in [65]. In particular, the Padé series expansion discussed in [65,112] is fairly

well-known. It can be shown that the matrix partitions in (B4) represent the following matrices:

$$F_1(t_2-t_1) = e^{A(t_2-t_1)}, \tag{B5}$$

$$F_2(t_2-t_1) = e^{C(t_2-t_1)}, \tag{B6}$$

$$F_3(t_2-t_1) = e^{E(t_2-t_1)}, \tag{B7}$$

$$G_1(t_2-t_1) = \int_{t_1}^{t_2} e^{A(t_2-\tau)} B e^{C(\tau-t_1)} d\tau, \tag{B8}$$

$$G_2(t_2-t_1) = \int_{t_1}^{t_2} e^{C(t_2-\tau)} D e^{E(\tau-t_1)} d\tau, \tag{B9}$$

and

$$H_1(t_2-t_1) = \int_{t_1}^{t_2} e^{A(t_2-\tau)} B \int_{t_1}^{\tau} e^{C(\tau-\varepsilon)} D e^{E(\varepsilon-t_1)} d\varepsilon \, d\tau. \tag{B10}$$

The desired integrals can then be identified as

$$G(A,B,C,t_2,t_1) = G_1(t_2-t_1), \tag{B11}$$

and

$$H(A,B,C,D,E,t_2,t_1) = H_1(t_2-t_1). \tag{B12}$$

Since the matrix exponential of (B4) satisfies the semi-group property

$$e^{\Pi(t_2-t_o)} = e^{\Pi(t_2-t_1)}e^{\Pi(t_1-t_o)} , \qquad (B13)$$

one can easily obtain the propagation equations for the partitions of (B4) by performing the partitioned matrix multiplication on

$$e^{\Pi(t+\Delta t)} = e^{\Pi\Delta t}e^{\Pi t} , \qquad (B14)$$

leading to

$$F_i(t+\Delta t) = F_i(\Delta t)F_i(t) , \qquad (i=1,2,3) , \quad (B15)$$

$$G_i(t+\Delta t) = F_i(\Delta t)G_i(t) + G_i(\Delta t)F_{i+1}(t), \qquad (i=1,2) , \quad (B16)$$

and

$$H_i(t+\Delta t) = F_i(\Delta t)H_i(t) + G_i(\Delta t)G_{i+1}(t) + H_i(\Delta t)F_{i+2}(t), \ (i=1) . \quad (B17)$$

Propagation of the matrix partitions via (B15) through (B17) involves matrix multiplications which require $O(n^3)$ flops per time-step, where n is an average row or column dimension of the matrix partitions.

When products of F_i, G_i, and H_i with constant vectors are to be propagated, there is a more practical propagation method than the matrix propagation and subsequent multiplication with the constant vectors. Let us denote the matrix vector products by

$$x_{Fi}(t) = F_i(t)u_i , \qquad (i=1,2,3) , \quad (B18)$$

$$x_{Gi}(t) = G_i(t)u_{i+1} , \qquad (i=1,2) , \quad (B19)$$

and

$$x_{Hi}(t) = H_i(t)u_{i+2} , \qquad (i=1) , \quad (B20)$$

where u_i are constant vectors. Postmultiplication of (B15) by u_i and substitution of (B18) gives

$$x_{Fi}(t+\Delta t) = F_i(\Delta t)x_{Fi}(t) , \qquad (i=1,2,3). \quad (B21)$$

Postmultiplication of (B16) by u_{i+1} and substitution of (B18) and (B19) yields

$$x_{Gi}(t+\Delta t) = F_i(\Delta t)x_{Gi}(t) + G_i(\Delta t)x_{Fi+1}(t) , \qquad (i=1,2), \quad (B22)$$

Postmultiplication of (B17) by u_{i+2} and substitution of (B18) through (B20) leads to

$$x_{Hi}(t+\Delta t) = F_i(\Delta t)x_{Hi}(t) + G_i(\Delta t)x_{Gi+1}(t) + H_i(\Delta t)x_{Fi+2}(t), \quad (i=1). \quad (B23)$$

Equations (B21) through (B23) represent the propagation of the matrix-vector products of (B18) through (B20). Since the propagation equations involve only matrix-vector multiplication, the propagation requires only $O(n^2)$ flops per time-step.

The above derivations are for the calculation of both $G(A,B,C,t_2,t_1)$ and $H(A,B,C,D,E,t_2,t_1)$ and their products with constant vectors. When one is only interested in $G(A,B,C,t_2,t_1)$ and its products with constant vectors, only the upper left partition of Π and $e^{\Pi(t_2-t_1)}$ need be used. That is,

$$\Pi = \begin{bmatrix} A & B \\ 0 & C \end{bmatrix} , \qquad (B24)$$

and

$$e^{\Pi(t_2-t_1)} = \begin{bmatrix} F_1(t_2-t_1) & G_1(t_2-t_1) \\ 0 & F_2(t_2-t_1) \end{bmatrix} . \qquad \text{(B25)}$$

If one were only interested in propagating matrix-vector products of the form $G(A,B,C,t_2,t_1)v$, where v is a constant vector, then one need only propagate $x_{G1}(t)$ and $x_{F2}(t)$ via (B21) and (B22) with $u_2=v$. Similarly, if one were only interested in propagating $H(A,B,C,D,E,t_2,t_1)v$, then one only needs to propagate $x_{H1}(t)$, $x_{G2}(t)$ and $x_{F3}(t)$ via (B21) through (B23) with $u_3=v$.

APPENDIX C

CONTROL-SMOOTHING FORMULATION

The concept of control-derivative penalty was first introduced by Moore and Anderson [66], and later expanded on by Gupta [36-37]. The usual finite-time linear-quadratic formulation results in controls with discontinuities at the initial and final times. By introducing control-derivative penalties in the performance index, one is able to specify initial and final values for the controls and their derivatives. Therefore, by means of the control-derivative penalties, one is able to design smooth control profiles. For example, for a rest-to-rest re-orientation maneuver, one may set the torques and torque-rates to zero at the initial and final times. For a targeting maneuver, one may set the final control torque so that the spacecraft angular acceleration closely approximates the target angular acceleration.

The introduction of control-derivative penalties in the performance index is easily done by augmenting the state vector with the controls and control-rates:

$$\tilde{x} = [x^T \; u_o^T \; u_1^T]^T \; , \tag{C1}$$

where x is the original state vector, u_o and u_1 are the control vector and its derivative, respectively, and $\tilde{x}$ is the augmented state. For simplicity, only the controls and their first time-derivatives are augmented for the development in this appendix. The generalization to higher time-derivatives is straightforward.

Given the original state differential equations of the form:

$$\dot{x}(t) = Ax(t) + Bu_o(t) \; , \tag{C2}$$

the augmented state dynamics can be written as

$$\dot{\tilde{x}} = \tilde{A}\tilde{x}(t) + \tilde{B}u_2(t) \tag{C3}$$

where

$$\tilde{A} = \begin{bmatrix} A & B & 0 \\ 0 & 0 & I \\ 0 & 0 & 0 \end{bmatrix}, \qquad \tilde{B} = \begin{bmatrix} 0 \\ 0 \\ I \end{bmatrix},$$

and u_2 is the second time derivative of the control vector. It is easily shown that the augmented system is controllable (or stabilizable) if the original system is controllable (or stabilizable).

A quadratic performance index for the augmented system can be written in the form

$$J = \frac{1}{2}\tilde{x}^T(t_f)P_f\tilde{x}(t_f) + \frac{1}{2}\int_{t_o}^{t_f}[\tilde{x}^T(t)\ u_2^T(t)]\begin{bmatrix} Q & N \\ N^T & R \end{bmatrix}\begin{Bmatrix} \tilde{x}(t) \\ u_2(t) \end{Bmatrix}dt , \tag{C4}$$

where $P_f = P_f^T \geq 0$ is the terminal state weight matrix, $Q = Q^T \geq 0$ is the state weight matrix, $R = R^T > 0$ is the weight matrix for the second control-derivative, and N is the cross weight matrix for the products of the augmented states and the second control-derivative. It has been known that for infinite-time steady-state optimal control problems, the inclusion of the cross weighting matrix N allows the movement of certain closed-loop eigenvalues in pole-placement algorithms when these eigenvalues are not movable by altering the Q and R matrices alone.

Gupta has shown that the minimization of frequency-shaped cost functionals requires the augmentation of the states and the cross-weight matrix for products of the augmented states and the higher derivatives of the control vector. For example, let us use a frequency domain cost functional of the form

$$J = \frac{1}{2}\int_{-\infty}^{\infty}[x^{*}(j\omega)\bar{Q}x(j\omega) + u_{o}^{*}(j\omega)|\bar{R}(j\omega)|^{2}u_{o}(j\omega)]d\omega \ , \tag{C5}$$

where $x(j\omega)$ and $u_o(j\omega)$ are the transforms of the states and controls, the asterisk * denotes the complex conjugate transpose, $\bar{Q}$ is a state weight matrix, and $\bar{R}(j\omega)$ is a frequency-dependent weight matrix on the controls. Let us choose a control weight matrix of the form [36]

$$\bar{R}(j\omega) = [(\frac{j\omega}{\omega_B})^2 + 1]I \ , \tag{C6}$$

so that frequencies of $u_o(j\omega)$ above ω_B are increasingly penalized. The cost functional of (C5) can be recast in an equivalent time domain form as follows. On setting $s = j\omega$, one can define an intermediate variable

$$u'(s) = \bar{R}(s)u(s) \ , \tag{C7}$$

which can be represented as the output of a dynamic system:

$$\begin{Bmatrix} \dot{u}_o(t) \\ \dot{u}_1(t) \end{Bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}\begin{Bmatrix} u_o(t) \\ u_1(t) \end{Bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix}u_2(t) \ , \tag{C8}$$

and

$$u'(t) = u_o(t) + u_2(t)/\omega_B^2 \ . \tag{C9}$$

Augmenting (C8) to (C2) leads to the augmented equation of (C3). Using Parseval's Theorem, one may write the time-domain equivalent of the cost functional of (C5) as

$$J = \frac{1}{2}\int_{0}^{\infty}[x^{T}(t)Qx(t) + u'^{T}(t)u'(t)]dt$$

$$= \frac{1}{2}\int_0^\infty [x^T(t)\ u_o^T(t)\ u_1^T(t)\ u_2^T(t)]\ W \begin{Bmatrix} x(t) \\ u_o(t) \\ u_1(t) \\ u_2(t) \end{Bmatrix} dt\ , \qquad \text{(C10)}$$

where

$$W = \begin{bmatrix} Q & 0 & 0 & 0 \\ 0 & I & 0 & I/\omega_B^2 \\ 0 & 0 & 0 & 0 \\ 0 & I/\omega_B^2 & 0 & I/\omega_B^4 \end{bmatrix} .$$

By choosing the weight matrices of (C4) for the finite-time control problem to be in the form of those in (C10) for the infinite-time problem, one can achieve the same type of roll-off in the control frequency spectrum for fixed time maneuvers as expected for the infinite-time problems.

For the performance index of (C4), it can be shown that the optimal control is given by [14]

$$u_2(t) = -R^{-1}[N^T + B^T P(t)]\tilde{x}(t)\ , \qquad \text{(C11)}$$

where

$$\dot{P} = -P\tilde{A} - \tilde{A}^T P + [P\tilde{B} + N]R^{-1}[N^T + \tilde{B}^T P] - Q\ ,$$

and

$$P(t_f) = P_f\ .$$

To simplify the numerical simulation of the above controller, one can recast the optimal control problem defined by (C3) and (C4) into the equivalent form:

$$J = \frac{1}{2}\tilde{x}^T(t_f)P_f\tilde{x}(t_f) + \frac{1}{2}\int_{t_o}^{t_f}[\tilde{x}^T(t) \quad u_2^T(t)]\begin{bmatrix} Q^* & 0 \\ 0 & R \end{bmatrix}\begin{Bmatrix} \tilde{x}(t) \\ u_2(t) \end{Bmatrix}dt$$

subject to

$$\dot{\tilde{x}} = A^*\tilde{x}(t) + \tilde{B}u_2(t) , \tag{C13}$$

where

$$A^* = \tilde{A} - \tilde{B}R^{-1}N^T ,$$

and

$$Q^* = Q - NR^{-1}N^T .$$

The control can then be expressed in the simplified form

$$u_2(t) = -R^{-1}\tilde{B}^TP^*(t)\tilde{x}(t) , \tag{C14}$$

where

$$\dot{P}^* = -P^*A^* - (A^*)^TP^* + P^*\tilde{B}R^{-1}\tilde{B}^TP^* - Q^* ,$$

and

$$P^*(t_f) = P_f .$$

APPENDIX D

SWEEP VARIABLE INITIAL CONDITIONS FOR THE TERMINAL CONTROLLER AND PERTURBATION FEEDBACK CONTROLLER

The sweep variable [63] initial conditions for the terminal controller of Section 2.3.2 and the perturbation feedback controller of Chapter 3 are computed by transforming the final conditions into initial conditions through the closed-form solutions for the sweep variable. Since the matrix $\bar{G}(t_f)$ for the terminal controller and the matrix $\bar{Q}(t_f)$ for the perturbation feedback are singular, it is necessary to use the matrix inversion lemma for expressions involving inverses of these matrices.

The matrix inversion lemma may be written as

$$[A - BC^{-1}D]^{-1} = A^{-1} - A^{-1}B[DA^{-1}B - C]^{-1}DA^{-1} , \qquad (D1)$$

where A and C are square matrices, and B and D may be rectangular matrices. In computing the initial conditions for the sweep variables, two types of expressions often appear in the derivations, namely $[A - BC^{-1}D]^{-1}BC^{-1}$ and $C^{-1} - C^{-1}D[A - BC^{-1}D]^{-1}BC^{-1}$. Let us apply the matrix inversion lemma to the first expression:

$$[A - BC^{-1}D]^{-1}BC^{-1} = A^{-1}BC^{-1} - A^{-1}B[DA^{-1}B - C]^{-1}DA^{-1}BC^{-1} . \qquad (D2)$$

Factoring out $A^{-1}B$ to the left and C^{-1} to the right, we get

$$[A - BC^{-1}D]^{-1}BC^{-1} = A^{-1}B[I - (DA^{-1}B-C)^{-1}DA^{-1}B]C^{-1} . \qquad (D3)$$

On writing the identity matrix in the above equation as $(DA^{-1}B-C)^{-1}$

$\times (DA^{-1}B-C)$, and simplifying, we obtain

$$[A - BC^{-1}D]^{-1}BC^{-1} = - A^{-1}B[DA^{-1}B - C]^{-1} . \quad (D4)$$

Substituting (D4) into the other often-occurring expression, we have

$$C^{-1} + C^{-1}D[A - BC^{-1}D]^{-1}BC^{-1} = C^{-1} - C^{-1}DA^{-1}B[DA^{-1}B - C]^{-1} . \quad (D5)$$

Factoring C^{-1} to the left leads to

$$C^{-1} + C^{-1}D[A - BC^{-1}D]^{-1}BC^{-1} = C^{-1}[I - DA^{-1}B(DA^{-1}B - C)^{-1}] . \quad (D6)$$

On writing the identity matrix in the above equation as $(DA^{-1}B - C)(DA^{-1}B - C)^{-1}$, and simplifying, we obtain

$$C^{-1} + C^{-1}D[A - BC^{-1}D]^{-1}BC^{-1} = - [DA^{-1}B - C]^{-1} . \quad (D7)$$

Terminal Controller

Using (D4) and (D7), we now proceed with the derivation of the initial conditions. For the terminal controller we make the following substitutions:

$$A = \bar{P} = \bar{P} - P_{ss} , \quad B = \bar{S}, \quad C = \bar{G}, \quad \text{and } D = \bar{S}^T . \quad (D8)$$

The final condition for Z(t) is given following (2.3.50) as

$$Z(t_f) = [P(t_f) - P_{ss}]^{-1} . \quad (D9)$$

Substitution of the expression for $P(t_f)$ following (2.3.46) leads to

$$Z(t_f) = [\tilde{P} - \bar{S}\bar{\bar{G}}^{-1}\bar{S}^T]^{-1} . \tag{D10}$$

Using the matrix inversion lemma of (D1), with (D8), one can re-write (D10) as

$$Z(t_f) = \tilde{P}^{-1} - \tilde{P}^{-1}\bar{S}[\bar{S}^T\tilde{P}^{-1}\bar{S} - \bar{G}]^{-1}\bar{S}^T\tilde{P}^{-1} . \tag{D11}$$

The initial condition for Z(t) is then obtained from the expression following (2.3.50):

$$Z(t_o) = Z_{ss} + e^{-\bar{A}(t_f-t_o)}[Z(t_f) - Z_{ss}]e^{-\bar{A}^T(t_f-t_o)} . \tag{D12}$$

The initial condition for S(t) is obtained by setting $t = t_f$ in (2.3.51) and solving for $S(t_o)$ to obtain

$$S(t_o) = Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}Z(t_f)S(t_f) . \tag{D13}$$

From the final condition for S(t) given after (2.3.46), the above expression becomes

$$S(t_o) = - Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}Z(t_f)\bar{S}\bar{\bar{G}}^{-1} . \tag{D14}$$

In order to eliminate the inverse of $\bar{G}$, we introduce the expression for $Z(t_f)$ from (D10), and consider the product

$$- Z(t_f)\bar{S}\bar{\bar{G}}^{-1} = - [\tilde{P} - \bar{S}\bar{\bar{G}}^{-1}\bar{S}^T]^{-1}\bar{S}\bar{\bar{G}}^{-1} , \tag{D15}$$

which is reduced by substitution of (D8) into (D4), leading to

$$- Z(t_f)\bar{S}\bar{G}^{-1} = \bar{P}^{-1}\bar{S}[\bar{S}^T\bar{P}^{-1}\bar{S} - \bar{G}]^{-1} . \quad (D16)$$

Substitution of (D16) into (D14) yields

$$S(t_o) = Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}\bar{P}^{-1}\bar{S}[\bar{S}^T\bar{P}^{-1}\bar{S} - \bar{G}]^{-1} . \quad (D17)$$

The initial condition for G(t) is obtained by setting $t = t_f$ in (2.3.52), and solving for $G(t_o)$ to get:

$$G(t_o) = S^T(t_o)Z(t_o)S(t_o) + G(t_f) - S^T(t_f)Z(t_f)S(t_f) . \quad (D18)$$

Using the expressions for $G(t_f)$ and $S(t_f)$ following (2.3.46), and the expression for $Z(t_f)$ from (D10), we re-write (D18) as

$$G(t_o) = S^T(t_o)Z(t_o)S(t_o) - \bar{G}^{-1} - \bar{G}^{-1}\bar{S}^T[\bar{P} - \bar{S}\bar{G}^{-1}\bar{S}^T]^{-1}\bar{S}\bar{G}^{-1} . \quad (D19)$$

The last two terms of the above equation may be simplified by substituting (D8) into (D7), yielding

$$G(t_o) = S^T(t_o)Z(t_o)S(t_o) + [\bar{S}^T\bar{P}^{-1}\bar{S} - \bar{G}]^{-1} . \quad (D20)$$

Perturbation Feedback Controller

For the perturbation feedback controller, we make the following substitutions for the matrix inversion lemma:

$$A = \tilde{S} = \bar{S} - S_{ss} , \quad B = \bar{R}, \quad C = \bar{Q}, \quad \text{and } D = \bar{R}^T . \quad (D21)$$

The final condition for Z(t) is shown in (3.4.19) as

$$Z(t_f) = [\tilde{S} - \bar{R}\bar{Q}^{-1}\bar{R}^T]^{-1} . \tag{D22}$$

To eliminate the inverse of $\bar{Q}$, which is singular, (D21) is substituted into (D1) to yield

$$Z(t_f) = \tilde{S}^{-1} - \tilde{S}^{-1}\bar{R}[\bar{R}^T\tilde{S}^{-1}\bar{R} - \bar{Q}]^{-1}\bar{R}^T\tilde{S}^{-1} . \tag{D23}$$

The initial condition for Z(t) may then be obtained from

$$Z(t_o) = Z_{ss} + e^{-\bar{A}(t_f-t_o)}[Z(t_f) - Z_{ss}]e^{-\bar{A}^T(t_f-t_o)} . \tag{D24}$$

The matrices $\hat{R}(t_o)$ and $\hat{R}(t_f)$ are related by (3.4.14) with $t = t_f$:

$$\hat{R}(t_o) = Z^{-1}(t_o)e^{-A(t_f-t_o)}Z(t_f)\hat{R}(t_f) . \tag{D25}$$

Using $\hat{R}(t_f)$ of (3.4.4) and $Z(t_f)$ of (3.4.19), we get

$$\hat{R}(t_o) = -Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}[\tilde{S} - \bar{R}\bar{Q}^{-1}\bar{R}^T]^{-1}\bar{R}\bar{Q}^{-1} . \tag{D26}$$

Substituting (D21) into (D4), and substituting the result into (D26) leads to the desired initial condition:

$$\hat{R}(t_o) = Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}\tilde{S}^{-1}\bar{R}[\bar{R}^T\tilde{S}^{-1}\bar{R} - \bar{Q}]^{-1} . \tag{D27}$$

For $\hat{m}(t)$, the initial condition is related to the terminal condition via (3.4.15) with $t = t_f$:

$$\hat{m}(t_o) = Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}Z(t_f)\hat{m}(t_f) . \tag{D28}$$

Using $\hat{m}(t_f)$ of (3.4.4) and $Z(t_f)$ of (3.4.19), we get

$$\hat{m}(t_o) = Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}(Z(t_f)\bar{m} - [\bar{S} - \bar{\bar{R}}\bar{\bar{Q}}^{-1}\bar{\bar{R}}^T]^{-1}\bar{\bar{R}}\bar{\bar{Q}}^{-1}\bar{n}) , \quad (D29)$$

which, upon substitution of (D4), becomes

$$\hat{m}(t_o) = - Z^{-1}(t_o)e^{-\bar{A}(t_f-t_o)}(Z(t_f)\bar{m} + \bar{S}^{-1}\bar{R}[\bar{R}^T\bar{S}^{-1}\bar{R} - \bar{Q}]^{-1}\bar{n}) . \quad (D30)$$

The initial condition for $\hat{Q}(t)$ is related to its terminal condition via (3.4.16) with $t = t_f$:

$$\hat{Q}(t_o) = \hat{R}^T(t_o)Z(t_o)\hat{R}(t_o) + \hat{Q}(t_f) - \hat{R}^T(t_f)Z(t_f)\hat{R}(t_f) . \quad (D31)$$

Using $\hat{R}(t_f)$ and $\hat{Q}(t_f)$ of (3.4.4) and $Z(t_f)$ of (3.4.19), we get

$$\hat{Q}(t_o) = \hat{R}^T(t_o)Z(t_o)\hat{R}(t_o) - \bar{Q}^{-1} - \bar{Q}^{-1}R^T[\bar{S} - \bar{\bar{R}}\bar{\bar{Q}}^{-1}\bar{\bar{R}}^T]^{-1}\bar{\bar{R}}\bar{\bar{Q}}^{-1} . \quad (D32)$$

For the last two terms of the above expression, substitution of (D21) into (D7), and the result into (D32) leads to the desired initial condition:

$$\hat{Q}(t_o) = \hat{R}^T(t_o)Z(t_o)\hat{R}(t_o) + [\bar{R}^T\bar{S}^{-1}\bar{R} - \bar{Q}]^{-1} . \quad (D33)$$

From (3.4.17), with $t = t_f$, we get an expression for $\hat{n}(t_o)$ in terms of $\hat{n}(t_f)$:

$$\hat{n}(t_o) = \hat{R}^T(t_o)Z(t_o)\hat{m}(t_o) + \hat{n}(t_f) - \hat{R}^T(t_f)Z(t_f)\hat{m}(t_f) . \quad (D34)$$

Using $\hat{n}(t_f)$, $\hat{R}(t_f)$, and $\hat{m}(t_f)$ of (3.4.4) and $Z(t_f)$ of (3.4.19), we get

$$\hat{n}(t_o) = \hat{R}(t_o)Z(t_o)\hat{m}(t_o) + (\bar{\bar{Q}}^{-1} + \bar{\bar{Q}}^{-1}\bar{\bar{R}}^T[\tilde{S} - \bar{\bar{R}}\bar{\bar{Q}}^{-1}\bar{\bar{R}}^T]^{-1}\bar{\bar{R}}\bar{\bar{Q}}^{-1})\bar{n}$$

$$- \bar{\bar{Q}}^{-1}R^T[\tilde{S} - \bar{\bar{R}}\bar{\bar{Q}}^{-1}\bar{\bar{R}}^T]^{-1}\bar{m} . \qquad (D35)$$

Using (D4) and (D7) with (D21), and simplifying, one can reduce the above expression to the form:

$$\hat{n}(t_o) = \hat{R}^T(t_o)Z(t_o)\hat{m}(t_o) + [\bar{\bar{R}}^T\tilde{S}^{-1}\bar{\bar{R}} - \bar{\bar{Q}}]^{-1}(\bar{\bar{R}}^T\tilde{S}^{-1}\bar{m} - \bar{n}) . \qquad (D36)$$

The initial value for the scalar $\hat{\alpha}(t)$ is given by (3.4.18) with $t = t_f$:

$$\hat{\alpha}(t_o) = \hat{m}^T(t_o)Z(t_o)\hat{m}(t_o) + \hat{\alpha}(t_f) - \hat{m}^T(t_f)Z(t_f)\hat{m}(t_f) . \qquad (D37)$$

Using the expressions for $\hat{\alpha}(t_f)$ and $\hat{m}(t_f)$ in (3.4.4), and $Z(t_f)$ in (3.4.19), we get

$$\hat{\alpha}(t_o) = \hat{m}^T(t_o)Z(t_o)\hat{m}(t_o) - \bar{\alpha} + \bar{m}^TZ(t_f)\hat{m}(t_f)$$

$$- \bar{n}^T(\bar{\bar{Q}}^{-1}\bar{n} + \bar{\bar{Q}}^{-1}\bar{\bar{R}}^T[\tilde{S} - \bar{\bar{R}}\bar{\bar{Q}}^{-1}\bar{\bar{R}}^T]^{-1}[\bar{\bar{R}}\bar{\bar{Q}}^{-1}\bar{n} - \bar{m}]) . \qquad (D38)$$

Observing that the term in curved brackets in (D38) is equal to the last two terms of (D35), and using (D28) for $Z(t_f)\hat{m}(t_f)$, we obtain

$$\hat{\alpha}(t_o) = [\bar{n}^T\hat{R}^T(t_o) + \hat{m}^T(t_o) + \bar{m}^Te^{\bar{A}(t_f-t_o)}]Z(t_o)\hat{m}(t_o) - \bar{n}^T\hat{n}(t_o) - \bar{\alpha} . \qquad (D39)$$

APPENDIX E

PROPAGATION OF THE RESIDUAL STATES FOR THE TRACKING/DISTURBANCE ACCOMMODATING CONTROLLER

The closed-form solution for the residual states for the tracking/disturbance accommodating controller is found to be

$$x_r(t) = e^{A_r(t-t_o)} x_r(t_o) - G(A_r, B_r D_1, -\bar{A}^T, t, t_o) a(t_o)$$

$$- G(A_r, B_r D_2, \bar{A}, t, t_o) b(t_o)$$

$$- [\, H(A_r, B_r D_1, -\bar{A}^T, K_2, \Omega, t, t_o) - H(A_r, B_r D_2, \bar{A}, K_1, \Omega, t, t_o) \,] s_o$$

$$+ [\, G(A_r, A_r, L, t, t_o) + H(A_r, B_r D_1, -\bar{A}^T, K_4, L, t, t_o)$$

$$- H(A_r, B_r D_2, \bar{A}, K_3, L, t, t_o) \,] \beta_o \,. \qquad \text{(E1)}$$

From Appendix B, five relevant matrices are identified as

$$\Pi_1 = \begin{bmatrix} A_r & B_r D_2 & 0 \\ 0 & \hat{A} & K_1 \\ 0 & 0 & \Omega \end{bmatrix}, \qquad \text{(E2)}$$

$$\Pi_2 = \begin{bmatrix} A_r & B_r D_1 & 0 \\ 0 & -\bar{A}^T & K_2 \\ 0 & 0 & \Omega \end{bmatrix}, \tag{E3}$$

$$\Pi_3 = \begin{bmatrix} A_r & B_r D_2 & 0 \\ 0 & \bar{A} & K_3 \\ 0 & 0 & L \end{bmatrix}, \tag{E4}$$

$$\Pi_r = \begin{bmatrix} A_r & B_r D_1 & 0 \\ 0 & -\bar{A}^T & K_4 \\ 0 & 0 & L \end{bmatrix}, \tag{E5}$$

and

$$\Pi_5 = \begin{bmatrix} A_r & \Lambda_r \\ 0 & L \end{bmatrix}. \tag{E6}$$

Let us define the partitions of the exponentials of the above matrices by

$$e^{\Pi_i t} = \begin{bmatrix} F_1^i(t) & G_1^i(t) & H_1^i(t) \\ 0 & F_2^i(t) & G_2^i(t) \\ 0 & 0 & F_3^i(t) \end{bmatrix}, \quad (i=1,\ldots,4), \tag{E7}$$

and

$$e^{\Pi_5 t} = \begin{bmatrix} F_1^5(t) & G_1^5(t) \\ 0 & F_2^5(t) \end{bmatrix}. \tag{E8}$$

Referring to (B5) through (B8), and observing the structure of the Π_i matrices in (E2) through (E6), one can make the following observations regarding partitions of the matrix exponentials:

$$F_1^1(t) = F_1^2(t) = F_1^3(t) = F_1^4(t) = F_1^5(t) = e^{A_r t} , \tag{E9}$$

$$F_2^1(t) = F_2^3(t) = e^{\bar{A}t} , \tag{E10}$$

$$F_2^2(t) = F_2^4(t) = e^{-\bar{A}^T t} , \tag{E11}$$

$$F_3^1(t) = F_3^2(t) = e^{\Omega t} , \tag{E12}$$

$$F_3^3(t) = F_3^4(t) = F_2^5(t) = e^{Lt} , \tag{E13}$$

$$G_1^1(t) = G_1^3(t) = G(A_r, B_r D_2, \bar{A}, t, 0) , \tag{E14}$$

and

$$G_1^2(t) = G_1^4(t) = G(A_r, B_r D_1, -\bar{A}^T, t, 0) . \tag{E15}$$

Let us now consider the propagation equations for each of the eight terms in (E1). The first term may be written as

$$x_{F1}^a(t) \triangleq e^{A_r(t-t_o)} x_r(t_o) = F_1^1(t) x_r(t_o) . \tag{E16}$$

From (B21), the propagation equation for $x_{F1}^a(t)$ is

$$x_{F1}^a(t+\Delta t) = F_1^1(t) x_{F1}^a . \tag{E17}$$

The second term in (E1) may be written as

$$x_{G1}^b(t) \triangleq G(A_r, B_r D_1, -\bar{A}^T, t, t_o) a(t_o) = G_1^2(t-t_o) a(t_o) , \tag{E18}$$

and propagated, using (B21) and (B22), by

$$x_{G1}^{b}(t+\Delta t) = F_1^2(\Delta t)x_{G1}^{b}(t) + G_1^2(\Delta t)x_{F2}^{b}(t) , \tag{E19}$$

where

$$x_{F2}^{b}(t) \triangleq F_2^2(t-t_o)a(t_o) , \tag{E20}$$

and

$$x_{F2}^{b}(t+\Delta t) = F_2^2(\Delta t)x_{F2}^{b}(t) . \tag{E21}$$

Similarly, the third term in (E1) may be written as

$$x_{G1}^{c}(t) \triangleq G(A_r,B_rD_2,\bar{A},t,t_o)b(t_o) = G_1^1(t-t_o)b(t_o) , \tag{E22}$$

and propagated by

$$x_{G1}^{c}(t+\Delta t) = F_1^1(\Delta t)x_{G1}^{c}(t) + G_1^1(\Delta t)x_{F2}^{c}(t) , \tag{E23}$$

where

$$x_{F2}^{c}(t) \triangleq F_2^1(t-t_o)b(t_o) , \tag{E24}$$

and

$$x_{F2}^{c}(t+\Delta t) = F_2^1(\Delta t)x_{F2}^{c}(t) . \tag{E25}$$

The fourth term in (E1) may be written as

$$x_{H1}^{d}(t) \triangleq H(A_r,B_rD_1,-\bar{A}^T,K_2,\Omega,t,t_o)s_o = H_1^2(t-t_o)s_o , \tag{E26}$$

and propagated, using (B21) through (B23), by

$$x_{H1}^{d}(t+\Delta t) = F_1^2(\Delta t)x_{H1}^{d}(t) + G_1^2(\Delta t)x_{G2}^{d}(t) + H_1^2(\Delta t)x_{F3}^{d}(t) , \tag{E27}$$

where

$$x^d_{G2}(t) \triangleq G^2_2(t-t_o)s_o \ , \tag{E28}$$

$$x^d_{G2}(t+\Delta t) = F^2_2(\Delta t)x^d_{G2}(t) + G^2_2(\Delta t)x^d_{F3}(t) \ , \tag{E29}$$

$$x^d_{F3}(t) \triangleq F^2_3(t-t_o)s_o \ , \tag{E30}$$

and

$$x^d_{F3}(t+\Delta t) = F^2_3(\Delta t)x^d_{F3}(t) \ , \tag{E31}$$

The fifth term in (E1) may be written as

$$x^e_{H1}(t) \triangleq H(A_r, B_r D_2, \bar{A}, K_1, \Omega, t, t_o)s_o = H^1_1(t-t_o)s_o \ , \tag{E32}$$

and propagated, using (B21) through (B23), by

$$x^e_{H1}(t+\Delta t) = F^1_1(\Delta t)x^e_{H1}(t) + G^1_1(\Delta t)x^e_{G2}(t) + H^1_1(\Delta t)x^e_{F3}(t) \ , \tag{E33}$$

where

$$x^e_{G2}(t) \triangleq G^1_2(t-t_o)s_o \ , \tag{E34}$$

$$x^e_{G2}(t+\Delta t) = F^1_2(\Delta t)x^e_{G2}(t) + G^1_2(\Delta t)x^e_{F3}(t) \ , \tag{E35}$$

$$x^e_{F3}(t) \triangleq F^1_3(t-t_o)s_o \ , \tag{E36}$$

and

$$x^e_{F3}(t+\Delta t) = F^1_3(\Delta t)x^e_{F3}(t) \ . \tag{E37}$$

The sixth term in (E1) may be written as

$$x_{G1}^{f}(t) \triangleq G(A_r,\Lambda_r,L,t,t_o)\beta_o = G_1^5(t-t_o)\beta_o \ , \tag{E38}$$

and propagated, using (B21) and (B22), by

$$x_{G1}^{f}(t+\Delta t) = F_1^5(\Delta t)x_{G1}^{f}(t) + G_1^5(\Delta t)x_{F2}^{f}(t) \ , \tag{E39}$$

where

$$x_{F2}^{f}(t) \triangleq F_2^5(t-t_o)\beta_o \ , \tag{E40}$$

and

$$x_{F2}^{f}(t+\Delta t) = F_2^5(\Delta t)x_{F2}^{f}(t) \ . \tag{E41}$$

The seventh term in (E1) may be written as

$$x_{H1}^{g}(t) \triangleq H(A_r,B_rD_1,-\bar{A}^T,K_4,L,t,t_o)\beta_o = H_1^4(t-t_o)\beta_o \ , \tag{E42}$$

and propagated, using (B21) through (B23), by

$$x_{H1}^{g}(t+\Delta t) = F_1^4(\Delta t)x_{H1}^{g}(t) + G_1^4(\Delta t)x_{G2}^{g}(t) + H_1^4(\Delta t)x_{F3}^{g}(t) \ , \tag{E43}$$

where

$$x_{G2}^{g}(t) \triangleq G_2^4(t-t_o)\beta_o \ , \tag{E44}$$

$$x_{G2}^{g}(t+\Delta t) = F_2^4(\Delta t)x_{G2}^{g}(t) + G_2^4(\Delta t)x_{F3}^{g}(t) \ , \tag{E45}$$

$$x_{F3}^{g}(t) \triangleq F_3^4(t-t_o)\beta_o \ , \tag{E46}$$

and

$$x^{g}_{F3}(t+\Delta t) = F^{4}_{3}(\Delta t)x^{g}_{F3}(t). \tag{E47}$$

Finally, the eighth term in (E1) may be written as

$$x^{g}_{H1}(t) \triangleq H(A_r,B_rD_2,\bar{A},K_3,L,t,t_o)\beta_o = H^{3}_{1}(t-t_o)\beta_o , \tag{E48}$$

and propagated, using (B21) through (B23), by

$$x^{h}_{H1}(t+\Delta t) = F^{3}_{1}(\Delta t)x^{h}_{H1}(t) + G^{3}_{1}(\Delta t)x^{h}_{G2}(t) + H^{3}_{1}(\Delta t)x^{h}_{F3}(t) , \tag{E49}$$

where

$$x^{h}_{G2}(t) \triangleq G^{3}_{2}(t-t_o)\beta_o , \tag{E50}$$

$$x^{h}_{G2}(t+\Delta t) = F^{3}_{2}(\Delta t)x^{h}_{G2}(t) + G^{3}_{2}(\Delta t)x^{h}_{F3}(t) , \tag{E51}$$

$$x^{h}_{F3}(t) \triangleq F^{3}_{3}(t-t_o)\beta_o , \tag{E52}$$

and

$$x^{h}_{F3}(t+\Delta t) = F^{3}_{3}(\Delta t)x^{h}_{F3}(t) . \tag{E53}$$

In terms of the new variables, the residual state solution may be expressed as

$$x_r(t) = x^{a}_{F1}(t) - x^{b}_{G1}(t) - x^{c}_{G1}(t) - x^{d}_{H1}(t) + x^{e}_{H1}(t) + x^{f}_{G1}(t)$$

$$+ x^{g}_{H1}(t) - x^{h}_{H1}(t) . \tag{E54}$$

Replacing t by t+Δt in the above equation, and using the propagation equations for each of the terms on the right hand side, one can obtain the propagation equation for $x_r(t)$.

Let us proceed in several steps. First, we collect terms on the right hand sides of the relevant propagation equations which involve matrices of the form $F(\Delta t)$. That is, from (E17), (E19), (E23), (E27), (E33), (E39), (E43), and (E49), we extract

$$x_F = F_1^1(\Delta t)x_{F1}^a(t) - F_1^2(\Delta t)x_{G1}^b(t) - F_1^1(\Delta t)x_{G1}^c(t) - F_1^2(\Delta t)x_{H1}^d(t)$$

$$+ F_1^1(\Delta t)x_{H1}^e(t) + F_1^5(\Delta t)x_{G1}^f(t) + F_1^4(\Delta t)x_{H1}^g(t) - F_1^3(\Delta t)x_{H1}^h(t) . \quad (E55)$$

From (E9) and (E54), the above expression becomes

$$x_F = e^{A_r \Delta t} x_r(t) . \quad (E56)$$

Next, we collect terms of the form $G(\Delta t)$ appearing on the right hand sides of the propagation equations for the individual terms in (E54). That is, from (E19), (E23), (E27), (E33), (E39), (E43), and (E49), we extract

$$x_G = - G_1^2(\Delta t)x_{F2}^b(t) - G_1^1(\Delta t)x_{F2}^c(t) - G_1^2(\Delta t)x_{G2}^d(t)$$

$$+ G_1^1(\Delta t)x_{G2}^e(t) + G_1^5(\Delta t)x_{F2}^f(t) + G_1^4(\Delta t)x_{G2}^g(t)$$

$$- G_1^3(\Delta t)x_{G2}^h(t) . \quad (E57)$$

Using (E14) and (E15), one can simplify the above expression as

$$x_G = - G_1^2(\Delta t)\bar{a}(t) - G_1^1(\Delta t)\bar{b}(t) + G_1^5(\Delta t)x_{F2}^f(t) , \quad (E58)$$

where

$$\bar{a}(t) = x_{F2}^{b}(t) + x_{G2}^{d}(t) - x_{G2}^{g}(t) , \tag{E59}$$

and

$$\bar{b}(t) = x_{F2}^{c}(t) - x_{G2}^{e}(t) + x_{G2}^{h}(t) . \tag{E60}$$

By comparing the initial conditions and propagation equations for $\bar{a}(t)$ and $\bar{b}(t)$ above with those for a(t) and b(t) of (2.3.127), it can be shown that $\bar{a}(t) = a(t)$, and $\bar{b}(t) = b(t)$. From (E20), (E28), and (E44), and using

$$F_i^j(0) = I, \qquad G_j^i(0) = 0 , \tag{E61}$$

for any i, j, one can show that the initial condition for $\bar{a}(t)$ is

$$\bar{a}(t_o) = a(t_o) . \tag{E62}$$

Similarly, from (E24), (E34), and (E50), the initial condition for $\bar{b}(t)$ is found to be

$$\bar{b}(t_o) = b(t_o) . \tag{E63}$$

To obtain the propagation equation for $\bar{a}(t)$, (E21), (E29), and (E45) are combined to give

$$\bar{a}(t+\Delta t) = e^{-\bar{A}^T \Delta t}\bar{a}(t) + G_2^2(\Delta t)s(t) - G_2^4(\Delta t)\beta(t) , \tag{E64}$$

where we have used (E11); and the fact that

$$x_{F3}^{d}(t) = e^{\Omega(t-t_o)}s_o = s(t) , \tag{E65}$$

from (2.3.90), (E12), and (E30); and

$$x_{F3}^{g}(t) = e^{L(t-t_o)}\beta_o = \beta(t) , \tag{E66}$$

from (2.3.86), (E13), and (E46). From (B9), (E3), and (E5), $G_2^2(\Delta t)$ and $G_2^4(\Delta t)$ may be identified as

$$G_2^2(\Delta t) = G(-\bar{A}^T, K_2, \Omega, \Delta t, 0) , \tag{E67}$$

and

$$G_2^4(\Delta t) = G(-\bar{A}^T, K_4, L, \Delta t, 0) . \tag{E68}$$

Comparison of (E64), (E67), and (E68) with (2.3.128), and the expressions for C_2 and C_4 following (2.3.126) shows that the propagation equation for $\bar{a}(t)$ is the same as that for $a(t)$. Since the initial conditions and propagation equations (they are actually difference equations) are the same, the variables $a(t)$ and $\bar{a}(t)$ are identical to each other. Similarly, the propagation equation for $\bar{b}(t)$ is obtained by combining (E25), (E35), and (E51) to give

$$\bar{b}(t+\Delta t) = e^{\bar{A}\Delta t}\bar{b}(t) + G_2^1(\Delta t)s(t) + G_2^3(\Delta t)\beta(t) , \tag{E69}$$

where we have used (E10), and the fact that

$$x_{F3}^{e}(t) = e^{\Omega(t-t_o)}s_o = s(t) , \tag{E70}$$

and

$$x_{F3}^{h}(t) = e^{L(t-t_o)}\beta_o = \beta(t) . \tag{E71}$$

From (B9), (E2), and (E4), $G_2^1(\Delta t)$ and $G_2^3(\Delta t)$ may be identified as

$$G_2^1(\Delta t) = G(\bar{A}, K_1, \Omega, \Delta t, 0) , \tag{E72}$$

and

$$G_2^3(\Delta t) = G(-\bar{A}^T, K_3, L, \Delta t, 0) . \tag{E73}$$

Comparison of (E69), (E72), and (E73) with (2.3.129), and the expressions for C_1 and C_3 following (2.3.126) shows that the propagation equation for $\bar{b}(t)$ is the same as that for $b(t)$. Since the initial conditions and the propagation equations are the same, the variables $b(t)$ and $\bar{b}(t)$ are identical to each other. The expression for x_G in (E58) may then be re-written as

$$x_G = - G_1^2(\Delta t)a(t) - G_1^1(\Delta t)b(t) + G_1^5(\Delta t)\beta(t) , \tag{E74}$$

where we have used the fact that

$$x_{F2}^f(t) = e^{L(t-t_o)}\beta_o = \beta(t) \tag{E75}$$

from (2.3.86), (E13), and (E40).

Finally, we collect terms of the form $H(\Delta t)$ appearing on the right hand sides of the propagation equations for the individual terms in (E54). That is, from (E27), (E33), (E43), and (E49), we extract

$$x_H = - H_1^2(\Delta t)x_{F3}^d(t) + H_1^1(\Delta t)x_{F3}^e(t) + H_1^4(\Delta t)x_{F3}^g(t)$$

$$- H_1^3(\Delta t)x_{F3}^h(t) . \tag{E76}$$

From (E65), (E66), (E70), and (E71), the above expression may be written as

$$x_H = \left(H_1^1(\Delta t) - H_1^2(\Delta)\right)s(t) + \left(H_1^4(\Delta t) - H_1^3(\Delta t)\right)\beta(t) . \qquad \text{(E77)}$$

The propagation equation for $x_r(t)$ may now be written as the sum of all terms on the right hand sides of the propagation equations for each of the terms in (E54). That is, from (E56), (E74), and (E77), we obtain

$$x_r(t+\Delta t) = x_F + x_G + x_H$$

$$= e^{A_r \Delta t} x_r(t) - D_3 a(t) - D_4 b(t) + B_5 s(t) + D_6 \beta(t) , \qquad \text{(E78)}$$

where

$$D_3 = G_1^2(\Delta t) = G(A_r, B_r D_1, -\bar{A}^T, \Delta t, 0) , \qquad \text{(E79)}$$

$$D_4 = G_1^1(\Delta t) = G(A_r, B_r D_1, \bar{A}, \Delta t, 0) , \qquad \text{(E80)}$$

$$D_5 = H_1^1(\Delta t) - H_1^2(\Delta t)$$

$$= H(A_r, B_r D_2, \bar{A}, K_1, \Omega, \Delta t, 0) - H(A_r, B_r D_1, -\bar{A}^T, K_2, \Omega, \Delta t, 0) , \qquad \text{(E81)}$$

and

$$D_6 = G_1^5(\Delta t) + H_1^4(\Delta t) - H_1^3(\Delta t)$$

$$= G(A_r, \Lambda_r, L, \Delta t, 0) + H(A_r, B_r D_1, -\bar{A}^T, K_4, L, \Delta t, 0)$$

$$- H(A_r, B_r D_2, \bar{A}, K_3, L, \Delta t, 0) . \qquad \text{(E82)}$$

In the above equations, (E79) and (E80) are obtained from (E15) and (E14), respectively; (E81) is obtained from (B10), using the matrices Π_1 and Π_2 of (E2) and (E3); and (E82) is obtained from (B8) and (B10), using the

matrices Π_3, Π_4, and Π_5 of (E4) through (E6).

APPENDIX F

TRANSFORMATION OF SWEEP VARIABLES FOR THE PERTURBATION FEEDBACK CONTROLLER

This appendix shows how the terminal conditions

$$\begin{Bmatrix} \delta\lambda_f \\ d\psi \\ d\Omega=0 \end{Bmatrix} = \begin{bmatrix} S & R & m \\ R^T & Q & n \\ m^T & n^T & \alpha \end{bmatrix} \begin{Bmatrix} \delta x_f \\ \nu \\ dt_f \end{Bmatrix} \tag{F1}$$

are manipulated into the form

$$\begin{Bmatrix} \delta\lambda_f \\ d\nu \\ dt_f \end{Bmatrix} = \begin{bmatrix} \hat{S} & \hat{R} & \hat{m} \\ \hat{R}^T & \hat{Q} & \hat{n} \\ \hat{m}^T & \hat{n}^T & \hat{\alpha} \end{bmatrix} \begin{Bmatrix} \delta x_f \\ -d\psi \\ -d\Omega=0 \end{Bmatrix} . \tag{F2}$$

First, the last row of (F1) is re-written as

$$dt_f = \frac{1}{\alpha} [d\Omega - m^T \delta x_f - n^T d\nu] . \tag{F3}$$

By using the expression for dt_f in (F3), the expressions for $\delta\lambda_f$ and $d\psi$ are re-written as

$$d\lambda_f = [S - \frac{mm^T}{\alpha}]\delta x_f + [R - \frac{mn^T}{\alpha}]d\nu + \frac{m}{\alpha} d\Omega , \tag{F4}$$

and

$$d\psi = [R^T - \frac{nm^T}{\alpha}]\delta x_f + [Q - \frac{nn^T}{\alpha}]d\nu + \frac{n}{\alpha} d\Omega . \tag{F5}$$

Solving (F5) for $d\nu$ leads to

$$d\nu = -[Q - \frac{nn^T}{\alpha}]^{-1}[R^T - \frac{nm^T}{\alpha}]\delta x_f + [Q - \frac{nn^T}{\alpha}]^{-1}d\psi - [Q - \frac{nn^T}{\alpha}]^{-1}\frac{n}{\alpha}\, d\Omega \ . \tag{F6}$$

On comparing (F6) with (F2), one obtains the expressions for the matrix coefficients $\hat{R}$, $\hat{Q}$, and $\hat{n}$:

$$\hat{R} = -[R - \frac{mn^T}{\alpha}][Q - \frac{nn^T}{\alpha}]^{-1} \ , \tag{F7}$$

$$\hat{Q} = -[Q - \frac{nn^T}{\alpha}]^{-1} \ , \tag{F8}$$

and

$$\hat{n} = [Q - \frac{nn^T}{\alpha}]^{-1}\frac{n}{\alpha} \ . \tag{F9}$$

Substituting the expression for $d\nu$ in (F6) into (F4) leads to

$$\delta\lambda_f = [S - \frac{mm^T}{\alpha} - (R - \frac{mn^T}{\alpha})(Q - \frac{nn^T}{\alpha})^{-1}(R^T - \frac{nm^T}{\alpha})]\delta x_f$$

$$+ [(R - \frac{mn^T}{\alpha})(Q - \frac{nn^T}{\alpha})^{-1}]d\psi - [(R - \frac{mn^T}{\alpha})(Q - \frac{nn^T}{\alpha})^{-1}\frac{n}{\alpha} - \frac{m}{\alpha}\, d\Omega \ . \tag{F10}$$

On comparing (F10) with (F2), one obtains the expressions for the matrix coefficients $\hat{S}$ and $\hat{m}$:

$$\hat{S} = S - \frac{mm^T}{\alpha} - [R - \frac{mn^T}{\alpha}][Q - \frac{nn^T}{\alpha}]^{-1}[R^T - \frac{nm^T}{\alpha}] \ , \tag{F11}$$

and

$$\hat{m} = [R - \frac{mn^T}{\alpha}][Q - \frac{nn^T}{\alpha}]^{-1} - \frac{m}{\alpha} . \tag{F12}$$

Finally, substituting (F6) into (F3), one gets

$$dt_f = [\frac{n^T}{\alpha}(Q - \frac{nn^T}{\alpha})^{-1}(R^T - \frac{nn^T}{\alpha}) - \frac{m^T}{\alpha}]\delta x_f - \frac{n^T}{\alpha}[Q - \frac{nn^T}{\alpha}]^{-1} d\psi$$

$$+ [\frac{n^T}{\alpha}(Q - \frac{nn^T}{\alpha})^{-1} \frac{n}{\alpha} + \frac{1}{\alpha}]d\Omega , \tag{F13}$$

which yields the expression for $\hat{\alpha}$:

$$\hat{\alpha} = - \frac{n^T}{\alpha} [Q - \frac{nn^T}{\alpha}]^{-1} \frac{n}{\alpha} - \frac{1}{\alpha} . \tag{F14}$$

APPENDIX G

PROPAGATION SCHEME FOR FEEDBACK GAINS OF THE PERTURBATION FEEDBACK CONTROLLER

This appendix shows the propagation scheme for efficiently computing the feedback gains at fixed time intervals. The Z(t) matrix, which appears in the solution of all of the feedback gains, is propagated by the following recursion:

$$Z(t+\Delta t) = C + e^{\bar{A}\Delta t} Z(t) e^{\bar{A}^T \Delta t} , \tag{G1}$$

where

$$C = Z_{ss} - e^{\bar{A}\Delta t} Z_{ss} e^{\bar{A}^T \Delta t} ,$$

Δt is the propagation time step, and the exponential matrix $e^{\bar{A}\Delta t}$ and C are computed only once.

On observing that $\hat{S}(t)$ always pre-multiplies a vector quantity, we can write (from (3.4.12))

$$\hat{S}(t+\Delta t)v = \hat{S}_{ss} v + Z^{-1}(t+\Delta t)v , \tag{G2}$$

where v represents any vector which may post-multiply $\hat{S}$, and $Z(t+\Delta t)$ is obtained using (G1). Note that (G2) does not represent a recursion formula, but rather indicates that $\hat{S}(t+\Delta t)v$ is obtained by a matrix-vector product and Gaussian elimination.

For $\hat{R}(t)$, we also observe that it is always post-multiplied by a vector, and therefore we can write (from (3.4.14))

$$\hat{R}(t+\Delta t)v = Z^{-1}(t+\Delta t)\tilde{R}(t+\Delta t)v \ , \tag{G3}$$

and

$$\tilde{R}(t+\Delta t) = e^{\bar{A}\Delta t}\tilde{R}(t) \ , \tag{G4}$$

where

$$\tilde{R}(t_o) = Z(t_o)\hat{R}(t_o) \ ,$$

and v denotes any vector which may post-multiply $\hat{R}$. Equation (G4) represents a recursion, and (G3) represents Gaussian elimination on the vector product $\tilde{R}(t+\Delta t)v$. The propagation equations for the vector $\hat{m}(t)$ are similar to those for $\hat{R}(t)$:

$$\hat{m}(t+\Delta t) = Z^{-1}(t+\Delta t)\tilde{m}(t+\Delta t) \ , \tag{G5}$$

and

$$\tilde{m}(t+\Delta t) = e^{\bar{A}\Delta t}\tilde{m}(t) \ , \tag{G6}$$

where

$$\tilde{m}(t_o) = Z(t_o)\hat{m}(t_o) \ .$$

On observing that $\tilde{R} = Z\hat{R}$ in (G3), we may re-write (3.4.17) for $\hat{n}(t)$ as

$$\hat{n}(t+\Delta t) = \tilde{R}^T(t+\Delta t)\hat{m}(t+\Delta t) + \hat{n}_c \ , \tag{G7}$$

where

$$\hat{n}_c = \hat{n}(t_o) - \hat{R}^T(t_o)Z(t_o)\hat{m}(t_o) \ .$$

Note that $\hat{n}_c$ is computed only once, and that (G7) represents a matrix-vector product and a vector addition.

It should be noted that in all of the above propagation equations, the inverse of $Z(t)$ is never computed explicitly. It should also be noted that for $t = t_f^N$ the gains become very large, and hence any slight numerical errors may cause divergence near the final time. To prevent this, one can replace $Q = 0$ of (3.4.2) by $Q = -\varepsilon_1 I$, where ε_1 is a small positive number. For similar reasons, the variable α is replaced by $(1+\varepsilon_2)\alpha$, where ε_2 is a small positive number. The modified Q matrix and α are then used, along with the definitions of S, R, m and n of (3.4.2), the $\bar{S}$, $\bar{R}$, $\bar{m}$, $\bar{Q}$, $\bar{n}$, and $\bar{\alpha}$ of (3.4.4), and (3.4.21) through (3.4.26) to compute the modified initial conditions for $\hat{S}(t)$, $\hat{R}(t)$, $\hat{m}(t)$, $\hat{Q}(t)$, $\hat{n}(t)$, and $\hat{\alpha}(t)$.

APPENDIX H

SCALING OF MODE SHAPES DUE TO CHANGES IN MASS AND LENGTH UNITS

Let us assume an equation of motion of the form

$$M\ddot{x} + Kx = w \,, \tag{H1}$$

where

$$M = \begin{bmatrix} m & s \\ s^T & J \end{bmatrix}, \quad x = \begin{Bmatrix} u \\ \theta \end{Bmatrix}, \quad w = \begin{Bmatrix} f \\ \tau \end{Bmatrix},$$

m, s, and J are matrices of masses, static mass moments and rotational inertia; u and θ are vectors of translational and rotational displacements, and f and τ are vectors of forces and torques. Assuming that mass is scaled by α, and distance scaled by β due to a change of units of mass and length, the variables in the equation of motion are scaled accordingly:

$$\tilde{M} = \begin{bmatrix} \alpha m & \alpha\beta s \\ \alpha\beta s^T & \alpha\beta^2 J \end{bmatrix} = GMG, \quad G = \begin{bmatrix} \sqrt{\alpha} I & 0 \\ 0 & \beta\sqrt{\alpha} I \end{bmatrix}, \tag{H2}$$

$$\tilde{K} = GKG \,, \tag{H3}$$

$$\tilde{x} = \begin{Bmatrix} \beta u \\ \theta \end{Bmatrix} = Fx, \quad F = \begin{bmatrix} \beta I & 0 \\ 0 & I \end{bmatrix}, \tag{H4}$$

$$\tilde{w} = \begin{Bmatrix} \alpha\beta f \\ \alpha\beta^2 \tau \end{Bmatrix} = GGFw \,, \tag{H5}$$

where the tilde superscript indicates variables in the new units. The new

equation of motion is then

$$\tilde{M}\ddot{\tilde{x}} + \tilde{K}\tilde{x} = \tilde{w} . \tag{H6}$$

Let us now perform modal transformations on the two equations of motion. For the original equations, we have

$$x = \Phi\xi , \tag{H7}$$

$$\Phi^T M\Phi = [M^*], \quad \Phi^T K\Phi = [M^*\omega^2] , \tag{H8}$$

and

$$[M^*]\ddot{\xi} + [M^*\omega^2]\xi = \Phi^T w , \tag{H9}$$

where $[M^*]$ and $[M^*\omega^2]$ are the diagonal modal mass and modal stiffness matrices, respectively. For the equation of motion in the new units, we have

$$\tilde{x} = \tilde{\Phi}\tilde{\xi} , \tag{H10}$$

$$\tilde{\Phi}^T\tilde{M}\tilde{\Phi} = [\tilde{M}^*] , \quad \tilde{\Phi}^T\tilde{K}\tilde{\Phi} = [\tilde{M}^*\omega^2] , \tag{H11}$$

and

$$[\tilde{M}^*]\ddot{\tilde{\xi}} + [\tilde{M}^*\omega^2]\tilde{\xi} = \tilde{\Phi}^T\tilde{w} . \tag{H12}$$

Let the new eigenvector matrix and modal amplitude vector be related to the original eigenvector matrix and modal amplitude vector as follows:

$$\tilde{\Phi} = H\Phi N , \tag{H13}$$

and

$$\tilde{\xi} = L\xi \ , \tag{H14}$$

where H, N, and L are positive definite diagonal matrices; in other words, we are permitting scaling but not linear combinations of rows or columns.

Combining (H4), (H7), (H10), (H13), and (H14), we get

$$\tilde{x} = H\Phi NL\xi = F\Phi\xi \ , \tag{H15}$$

which implies

$$H\Phi NL = F\Phi \ . \tag{H16}$$

The absence of post-multiplication on the right hand side of (H16) indicates that there is no columnwise manipulation of Φ in that equation. In order for (H16) to hold, one therefore must require that

$$NL = kI \ , \tag{H17}$$

where k is a constant factor to be determined.

Substituting (H17) into (H16) yields

$$H = \frac{1}{k} F \ . \tag{H18}$$

Using (H11), (H13), and (H18), we observe that the new modal mass matrix can be expressed in the original variables as

$$[\tilde{M}^*] = \frac{1}{k^2} N\Phi^T FGMGF\Phi N \ . \tag{H19}$$

From (H2) and (H4), we observe that the product GF is simply equal to $\beta\sqrt{\alpha}I$. Equation (H19) thus becomes

$$[\tilde{M}^*] = \frac{1}{k^2}\alpha\beta^2 N[M^*]N \ , \tag{H20}$$

which leads to

$$N = \frac{k}{\beta\sqrt{\alpha}}\left[[\tilde{M}^*][M^*]^{-1}\right]^{1/2} \ . \tag{H21}$$

Substituting (H18) and (H21) into (H13) yields

$$\tilde{\Phi} = G^{-1}\Phi\left[[\tilde{M}^*][M^*]^{-1}\right]^{1/2} \ , \tag{H22}$$

where again we have used the relationship $GF = \beta\sqrt{\alpha}I$. Similarly, using (H14), (H17) and (H21), we get

$$\tilde{\xi} = \beta\sqrt{\alpha}\left[[M^*][\tilde{M}^*]^{-1}\right]^{1/2}\xi \ . \tag{H23}$$

Note that the unknown constant k has dropped out of the problem. Equations (H22) and (H23) indicate that the appropriate dimensions for mode shapes and modal amplitudes are as follows:

a) For normalization schemes where the modal mass matrix is normalized to the identity matrix, we have

$$[M^*][\tilde{M}^*]^{-1} = I \ , \tag{H24}$$

which leads to

$$\tilde{\Phi} = G^{-1}\Phi \ , \tag{H25}$$

and

$$\tilde{\xi} = \beta\sqrt{\alpha}\xi \ . \tag{H26}$$

The appropriate units are then

(i) $(\text{mass}^{-1/2})$ for translational mode shapes,

(ii) $(\text{length}^{-1})(\text{mass}^{-1/2})$ for rotational mode shapes,

and

(iii) $(\text{length})(\text{mass}^{1/2})$ for modal amplitudes.

b) For normalization schemes where the modal mass matrix is normalized to the mass of the structure, we have

$$[M^*][\tilde{M}^*]^{-1} = \frac{1}{\alpha}I \ , \tag{H27}$$

which leads to

$$\tilde{\Phi} = \begin{bmatrix} I & 0 \\ 0 & \frac{1}{\beta}I \end{bmatrix}\Phi \ , \tag{H28}$$

and

$$\tilde{\xi} = \beta\xi \ . \tag{H29}$$

The appropriate units are then

(i) nondimensional translational mode shapes,

(ii) (length^{-1}) for rotational mode shapes,

and

(iii) (length) for modal amplitudes.

c) For normalization schemes where the modal mass matrix is normalized to a moment of inertia, we have

$$[M^*][\tilde{M}^*]^{-1} = \frac{1}{\alpha\beta^2}I \ ,$$

which leads to

$$\tilde{\Phi} = \begin{bmatrix} \beta & 0 \\ 0 & I \end{bmatrix} \Phi \ , \tag{H31}$$

and

$$\tilde{\xi} = \xi \ . \tag{H32}$$

The appropriate units are then

(i) (length) for translational mode shapes,

(ii) nondimensional rotational mode shapes,

and

(iii) nondimensional modal amplitudes.

APPENDIX I

PARTIAL DERIVATIVES OF STATE-COSTATE DIFFERENTIAL EQUATIONS FOR RIGID BODY MANEUVERS

For the continuation process of Section 4.3.5, the partials of the states with respect to the initial costates, i.e. $\partial\chi(t)/\partial\Lambda(t_o)$, must be obtained for $t = t_f$. This matrix is obtained by integration of its derivative from t_o to t_f. The derivative matrix is found by

$$\frac{d}{dt}\left[\frac{\partial\chi(t)}{\partial\Lambda(t_o)}\right] = \frac{\partial}{\partial\Lambda(t_o)}\left[\dot{\chi}(t)\right]$$

$$= \left[\frac{\partial\dot{\chi}(t)}{\partial\chi(t)}\right]\left[\frac{\partial\chi(t)}{\partial\Lambda(t_o)}\right] + \left[\frac{\partial\dot{\chi}(t)}{\partial\Lambda(t)}\right]\left[\frac{\partial\Lambda(t)}{\partial\Lambda(t_o)}\right] . \qquad (I1)$$

Thus the matrix $\partial\Lambda(t)/\partial\Lambda(t_o)$ must also be integrated along with $\partial\chi(t)/\partial\Lambda(t_o)$. The derivative for $\partial\Lambda(t)/\partial\Lambda(t_o)$ is similarly found to be

$$\frac{d}{dt}\left[\frac{\partial\Lambda(t)}{\partial\Lambda(t_o)}\right] = \frac{\partial}{\partial\Lambda(t_o)}\left[\dot{\Lambda}(t)\right]$$

$$= \left[\frac{\partial\dot{\Lambda}(t)}{\partial\chi(t)}\right]\left[\frac{\partial\chi(t)}{\partial\Lambda(t_o)}\right] + \left[\frac{\partial\dot{\Lambda}(t)}{\partial\Lambda(t)}\right]\left[\frac{\partial\Lambda(t)}{\partial\Lambda(t_o)}\right] . \qquad (I2)$$

The partials $\partial\chi(t)/\partial\Lambda(t_o)$ and $\partial\Lambda(t)/\partial\Lambda(t_o)$ are two partitions of the state transition matrix for the state-costate system. Thus, at the initial time, the boundary conditions are given by

$$\frac{\partial\chi(t_o)}{\partial\Lambda(t_o)} = 0 , \qquad \text{and} \qquad \frac{\partial\Lambda(t_o)}{\partial\Lambda(t_o)} = I . \qquad (I3)$$

On identification of the form of the state vector $\chi(t)$ and the costate vector $\Lambda(t)$ as

$$\chi(t) = [\beta^T \ \omega^T \ u_o^T \ u_1^T]^T ,$$

and

$$\Lambda(t) = [\gamma^T \ \lambda^T \ \mu_o^T \ \mu_1^T]^T , \tag{I4}$$

the elements of the coefficient matrices $\partial\dot{\chi}(t)/\partial\chi(t)$, $\partial\dot{\chi}(t)/\partial\Lambda(t)$, $\partial\dot{\Lambda}(t)/\partial\chi(t)$, and $\partial\dot{\Lambda}(t)/\partial\Lambda(t)$ of (I1) and (I2) are listed as follows:

Partials of $\dot{\beta}(t)$

$$\frac{\partial\dot{\beta}}{\partial\beta} = \Omega(\omega) = \frac{1}{2}\begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} ,$$

$$\frac{\partial\dot{\beta}}{\partial\omega} = B(\beta) = \frac{1}{2}\begin{bmatrix} -\beta_1 & -\beta_2 & -\beta_3 \\ \beta_o & -\beta_3 & \beta_2 \\ \beta_3 & \beta_o & -\beta_1 \\ -\beta_2 & \beta_1 & \beta_o \end{bmatrix} ,$$

$$\frac{\partial\dot{\beta}}{\partial u_o} = 0 ,$$

$$\frac{\partial\dot{\beta}}{\partial u_1} = 0 ,$$

$$\frac{\partial\dot{\beta}}{\partial\gamma} = 0 ,$$

$$\frac{\partial\dot{\beta}}{\partial\lambda} = 0 ,$$

$$\frac{\partial \dot{\beta}}{\partial \mu_o} = 0 ,$$

and

$$\frac{\partial \dot{\beta}}{\partial \mu_1} = 0 . \qquad \text{(I5)}$$

Partials of $\dot{\omega}(t)$

$$\frac{\partial \dot{\omega}}{\partial \beta} = 0 ,$$

$$\frac{\partial \dot{\omega}}{\partial \omega} = [I]^{-1}\{-[\tilde{\omega}][I] + \widetilde{[I\omega]}\} ,$$

$$\frac{\partial \dot{\omega}}{\partial u_o} = I , \qquad \text{(Identity)}$$

$$\frac{\partial \dot{\omega}}{\partial u_1} = 0 ,$$

$$\frac{\partial \dot{\omega}}{\partial \gamma} = 0 ,$$

$$\frac{\partial \dot{\omega}}{\partial \lambda} = 0 ,$$

$$\frac{\partial \dot{\omega}}{\partial \mu_o} = 0 ,$$

and

$$\frac{\partial \dot{\omega}}{\partial \mu_1} = 0 . \qquad \text{(I6)}$$

Partials of $\dot{u}_o(t)$

$$\frac{\partial \dot{u}_o}{\partial \beta} = 0 ,$$

$$\frac{\partial \dot{u}_o}{\partial \omega} = 0 ,$$

$$\frac{\partial \dot{u}_o}{\partial u_o} = 0 ,$$

$$\frac{\partial \dot{u}_o}{\partial u_1} = I , \qquad \text{(Identity)}$$

$$\frac{\partial \dot{u}_o}{\partial \gamma} = 0 ,$$

$$\frac{\partial \dot{u}_o}{\partial \lambda} = 0 ,$$

$$\frac{\partial \dot{u}_o}{\partial \mu_o} = 0 ,$$

and

$$\frac{\partial \dot{u}_o}{\partial \mu_1} = 0 . \qquad \text{(I7)}$$

Partials of $\dot{u}_1(t)$

$$\frac{\partial \dot{u}_1}{\partial \beta} = 0 ,$$

$$\frac{\partial \dot{u}_1}{\partial \omega} = 0 ,$$

$$\frac{\partial \dot{u}_1}{\partial u_o} = -\omega_B^2 I \ ,$$

$$\frac{\partial \dot{u}_1}{\partial u_1} = 0 \ ,$$

$$\frac{\partial \dot{u}_1}{\partial \gamma} = 0 \ ,$$

$$\frac{\partial \dot{u}_1}{\partial \lambda} = 0 \ ,$$

$$\frac{\partial \dot{u}_1}{\partial \mu_o} = 0 \ ,$$

and

$$\frac{\partial \dot{u}_1}{\partial \mu_1} = -\omega_B^4 I \ . \qquad (I8)$$

Partials of $\dot{\gamma}(t)$

$$\frac{\partial \dot{\gamma}}{\partial \beta} = 0 \ ,$$

$$\frac{\partial \dot{\gamma}}{\partial \omega} = \Gamma(\gamma) = \frac{1}{2}\begin{bmatrix} -\gamma_1 & -\gamma_2 & -\gamma_3 \\ \gamma_o & -\gamma_3 & \gamma_2 \\ \gamma_3 & \gamma_o & -\gamma_1 \\ -\gamma_2 & \gamma_1 & \gamma_o \end{bmatrix} ,$$

$$\frac{\partial \dot{\gamma}}{\partial u_o} = 0 \ ,$$

$$\frac{\partial \dot{\gamma}}{\partial u_1} = 0 ,$$

$$\frac{\partial \dot{\gamma}}{\partial \gamma} = \Omega(\omega) ,$$

$$\frac{\partial \dot{\gamma}}{\partial \lambda} = 0 ,$$

$$\frac{\partial \dot{\gamma}}{\partial \mu_o} = 0 ,$$

and

$$\frac{\partial \dot{\gamma}}{\partial \mu_1} = 0 . \qquad (I9)$$

Partials of $\dot{\lambda}(t)$

$$\frac{\partial \dot{\lambda}}{\partial \beta} = \Gamma^T(\gamma) ;$$

$$\frac{\partial \dot{\lambda}}{\partial \omega} = - Q - \frac{\partial}{\partial \omega} \left[\left(\frac{\partial G}{\partial \omega}\right)^T \lambda \right] ;$$

where

$$\left[\frac{\partial \dot{\lambda}}{\partial \omega}\right]_{11} = - Q_{11} + 2I_{xz}b - 2I_{xy}c ,$$

$$\left[\frac{\partial \dot{\lambda}}{\partial \omega}\right]_{12} = - Q_{12} - I_{xz}a + I_{yz}b + (I_{yy}-I_{xx})c ,$$

$$\left[\frac{\partial \dot{\lambda}}{\partial \omega}\right]_{13} = - Q_{13} + I_{xy}a + (I_{xx}-I_{zz})b - I_{yz}c ,$$

$$\left[\frac{\partial \dot{\lambda}}{\partial \omega}\right]_{21} = - Q_{21} - I_{xz}a + I_{yz}b + (I_{yy}-I_{xx})c ,$$

$$\left[\frac{\partial \dot{\lambda}}{\partial \omega}\right]_{22} = - Q_{22} - 2I_{xz}a + 2I_{xy}c ,$$

$$\left[\frac{\partial \dot{\lambda}}{\partial \omega}\right]_{23} = -Q_{23} + (I_{zz}-I_{yy})a - I_{xy}b + I_{xz}c \ ,$$

$$\left[\frac{\partial \dot{\lambda}}{\partial \omega}\right]_{31} = -Q_{31} + I_{xy}a + (I_{xx}-I_{zz})b - I_{yz}c \ ,$$

$$\left[\frac{\partial \dot{\lambda}}{\partial \omega}\right]_{32} = -Q_{32} + (I_{zz}-I_{yy})a - I_{xy}b + I_{xz}c \ ,$$

$$\left[\frac{\partial \dot{\lambda}}{\partial \omega}\right]_{33} = -Q_{33} + 2I_{yz}a - 2I_{xz}b \ ,$$

and

$$\begin{Bmatrix} a \\ b \\ c \end{Bmatrix} = [I]^{-1} \begin{Bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{Bmatrix} ;$$

$$\frac{\partial \dot{\lambda}}{\partial u_o} = 0 \ ;$$

$$\frac{\partial \dot{\lambda}}{\partial u_1} = 0 \ ;$$

$$\frac{\partial \dot{\lambda}}{\partial \gamma} = -B^T \ ;$$

$$\frac{\partial \dot{\lambda}}{\partial \lambda} = \{[\widetilde{I\omega}] - [I][\tilde{\omega}]\}[I]^{-1} \ ;$$

$$\frac{\partial \dot{\lambda}}{\partial \mu_o} = 0 \ ;$$

and

$$\frac{\partial \dot{\lambda}}{\partial \mu_1} = 0 \ . \qquad \text{(I10)}$$

Partials of $\dot{\mu}_o(t)$

$$\frac{\partial \dot{\mu}_o}{\partial \beta} = 0 ,$$

$$\frac{\partial \dot{\mu}_o}{\partial \omega} = 0 ,$$

$$\frac{\partial \dot{\mu}_o}{\partial u_o} = 0 ,$$

$$\frac{\partial \dot{\mu}_o}{\partial u_1} = 0 ,$$

$$\frac{\partial \dot{\mu}_o}{\partial \gamma} = 0 ,$$

$$\frac{\partial \dot{\mu}_o}{\partial \lambda} = -I , \qquad \text{(Identity)}$$

$$\frac{\partial \dot{\mu}_o}{\partial \mu_o} = 0 ,$$

and

$$\frac{\partial \dot{\mu}_o}{\partial \mu_1} = \omega_B^2 I . \qquad \text{(I11)}$$

Partials of $\dot{\mu}_1(t)$

$$\frac{\partial \dot{\mu}_1}{\partial \beta} = 0 ,$$

$$\frac{\partial \dot{\mu}_1}{\partial \omega} = 0 ,$$

$$\frac{\partial \dot{\mu}_1}{\partial u_o} = 0 ,$$

$$\frac{\partial \dot{\mu}_1}{\partial u_1} = 0 ,$$

$$\frac{\partial \dot{\mu}_1}{\partial \gamma} = 0 ,$$

$$\frac{\partial \dot{\mu}_1}{\partial \lambda} = 0 ,$$

$$\frac{\partial \dot{\mu}_1}{\partial \mu_o} = - I , \qquad \text{(Identity)}$$

and

$$\frac{\partial \dot{\mu}_1}{\partial \mu_1} = 0 . \qquad \text{(I12)}$$

The above analytical partial derivatives have been numerically verified by finite difference approximations.

VITA

The author was born in Hong Kong in 1960, and immigrated to the United States in 1965. He attended Illinois Institute of Technology from 1977 to 1978, and transferred to Massachusetts Institute of Technology, where he obtained his Bachelor of Science degree in 1981, and his Master of Science degree in 1982, both from the Department of Aeronautics and Astronautics. He began working toward his doctoral degree at Massachusetts Institute of Technology in 1983.

Publications by the author include articles in Journal of Guidance, Control, and Dynamics, the Journal of the Astronautical Sciences, Journal of Optimization Theory and Applications, and Journal of Dynamic Systems, Measurement, and Control, in addition to many conference papers and technical reports. The author is also reviewer for Journal of Guidance, Control, and Dynamics, the Journal of the Astronautical Sciences, and Journal of Optimization Theory and Applications.

Previous work experience includes half a year as a co-operative education student at NASA, Dryden Flight Research Center, and six years at the Charles Stark Draper Laboratory, Inc., under various titles. The author will join Cambridge Research, A Division of Photon Research Associates, Inc. in the fall of 1986 as staff scientist.