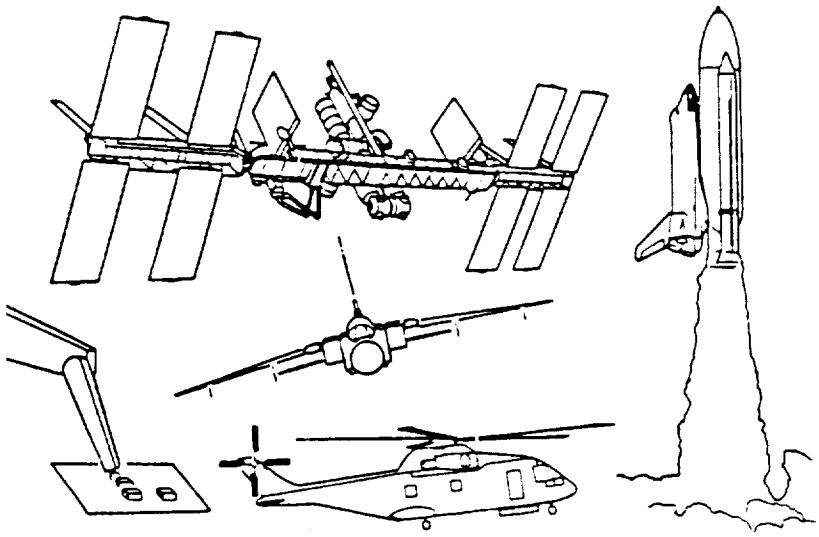
NASA Conference Publication 10065
Part 1

4th NASA Workshop on Computational Control of Flexible Aerospace Systems



Compiled by
Lawrence W. Taylor, Jr.
Langley Research Center
Hampton, Virginia

Proceedings of a workshop sponsored by the National Aeronautics and Space Administration and held at the Kingsmill Resort Williamsburg, Virginia July 11, 1990

MARCH 1991

NASA
National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

Foreword

The practice of modeling and controlling flexible aerospace systems grows in importance as the performance needed of active control systems increases. As the size of spacecraft increases and the demands of control systems become more exacting, the accuracy required of the models used for analysis also increases.

The increased complexity, the increased model accuracy, and the demands for more precise and higher control system performance result in an increased burden on the part of the analyst. Although this burden is somewhat alleviated by advances in software, there remains the pressure for assuring system stability and performance under conditions of plant uncertainty. Although robust considerations are included in many synthesis techniques, the price in terms of reduced system performance is often prohibitive.

Because similar difficulties and concerns are encountered for different applications, it is valuable to enhance the exchange of information with regard to aircraft, spacecraft and robotic applications. This is the fourth workshop in a series which has emphasized the computational aspects of controlling flexible aerospace systems. It is hoped that the reports contained in this proceedings will be useful to practicioners of modeling and controlling flexible systems.

Lawrence W. Taylor, Jr.
NASA Langley Research Center

TABLE of CONTENTS - Part 1

Table of Contents - Continued

Table of Contents - Continued

Table of Contents - Part 2*

* Published under separate cover

Table of Contents - Continued

Table of Contents - Continued

SPILLOVER, NONLINEARITY, & FLEXIBLE STRUCTURES

Robert W. Bass
Rockwell International Science Center, P.O. Box 1085, Thousand Oaks, CA 91358

Dean Zes
McDonnell Douglas Helicopter Co. 5000 McDowell Road, Mesa, AZ 85205

ABSTRACT

Many systems whose evolution in time is governed by Partial Differential Equations (PDEs) are linearized around a known equilibrium before Computer Aided Control Engineering (CACE) is considered. In this case there are infinitely many *independent* vibrational modes, and it is intuitively evident on physical grounds that infinitely many actuators would be needed in order to control *all* modes.

A more precise, general formulation of this grave difficulty (the "spillover" problem) is due to A.V. Balakrishnan [*Applied Functional Analysis*, Springer, 1981, p. 233]. Let the system's state vector x be an element of a separable Hilbert space $\mathcal{H}$ whose dimension is not finite; let A be a closed linear operator with domain dense in $\mathcal{H}$ which is the infinitesimal generator of a strongly continuous semi-group of transition operators $T(t)$ for non-negative times t; and let B denote a bounded linear operator acting on another separable Hilbert space $\mathcal{U}$ (the control space) with range in $\mathcal{H}$. Now consider the control problem $dx/dt = Ax + Bu$, with $x(0)$ given. Then according to Balakrishnan this system is not *exactly controllable* if B is compact.

A possible route to circumvention of this difficulty lies in leaving the PDE in its original *nonlinear* form, and adding the essentially finite-dimensional control action Bu prior to linearization. In many cases it can be shown that the nonlinearity couples the system's modes in such a manner that only a finite-dimensional subset of the modes is functionally independent, with the remaining higher-order modes nonlinearly dependent upon them. Hence control of *all* modes can be achieved by controlling only finitely many modes.

One possibly applicable technique is the Liapunov-Schmidt rigorous reduction of singular infinite-dimensional implicit function problems to finite-dimensional implicit function problems. Such a procedure was employed by Leon Lichtenstein in the 1930's to prove the existence of a solution of the Navier-Stokes equations for a sufficiently small time-interval $0 \le t < \varepsilon$.

Omitting details of Banach-space rigor, the formalities of this approach are as follows. Let $\mathcal{L}$ be a Fredholm operator with pseudo-inverse $\mathcal{K}$; then there exist idempotent projection operators $\mathcal{P} = \mathcal{I} - \mathcal{K}\mathcal{L}$, $\mathcal{Q} = \mathcal{I} - \mathcal{L}\mathcal{K}$ whose ranges are finite-dimensional and such that a NASC for (*) $\mathcal{L}x - \mathcal{F}(x) = 0$ is that (**) $x = \mathcal{P}x + \mathcal{K}\mathcal{F}(x)$ & $\mathcal{Q}\mathcal{F}(x) = 0$. Thus one may set $x = u + v$ where v satisfies the *auxiliary equation* $v = \mathcal{K}\mathcal{F}(u + v)$ and u the *bifurcation equations* [Verzweigungsgleichungen] $\mathcal{Q}\mathcal{F}(u + v) = 0$. Typically one may solve the auxiliary equation by (contraction) iterations to find $v = \mathcal{G}(u)$ where now v is infinite-dimensional but u is finite-dimensional and then insert the result into the (finitely many) bifurcation equations to define a finite-dimensional vector function $f(u) = \mathcal{Q}\mathcal{F}(u + \mathcal{G}(u))$ such that (*) is equivalent to $f(u) = 0$. In summary, a NASC for (*) is

$$(***)\quad x = u + \mathcal{G}(u),\quad \mathcal{L}u = 0,\quad \mathcal{P}\mathcal{G}(u) = 0,\quad \mathcal{P}u = u,\quad f(u) = 0.$$

As an illustration the auxiliary equation and bifurcation equations for the problem of *deflection of an in tension* ($a_0 > 0$) *EXTENSIBLE beam* ($a_1 > 0$) is considered, including *viscous damping* ($a_3 > 0$) and *Balakrishnan-Taylor damping* ($a_4 > 0$). Here

$$u_{tt} + a_2 \cdot u_{xxxx} = a_3 \cdot u_{xxt} + \left(a_0 + a_1 \cdot \int_0^L (u_x)^2 \cdot dx + a_4 \cdot \left[\int_0^L (u_x u_{xt}) \cdot dx \right]^{2(n+\beta)+1} \right) \cdot u_{xx},$$

$\forall\ t \ge 0$, $0 \le x \le L$. As the dimension N of the bifurcation equations increases, the result approaches an N-dimensional *truncated eigenexpansion* (provided that the initial deflections and their initial spatial and temporal rates of change are not too large).

Preface

The basic idea behind the present paper is simply:

> **SUGGESTION**
> **Don't linearize a PDE until *after* its reduction to a finite-dimensional ODE.**

This idea can be implemented by means of the following analytical procedure:

> **LIAPUNOV-SCHMIDT BIFURCATION EQUATIONS:**
> **A *rigorous* reduction of a singular *infinite*-dimensional implicit equation to the problem of an equivalent, merely *finite*-dimensional implicit equation.**

This suggestion is presented as a possible technique for circumvention of the famous "Spillover Problem."

Introduction

If the problem of control of a flexible structure is linearized before one considers the control aspects, then frequently it leads to an abstract problem in functional analysis of the type of the following system of ordinary differential equations:

$$\mathcal{S}: \qquad \frac{dx}{dt} = Ax + Bu, \qquad x(0) = x^0. \tag{1}$$

Here $x \in \mathcal{H}^\infty$ is an element of an infinite-dimensional state space taken to be a separable Hilbert space. Also $u \in \mathcal{U}$ is an element of the control space, taken to be another separable Hilbert space. We take A: $\mathcal{D} \to \mathcal{H}^\infty$ to denote a closed linear mapping of the dense linear-subspace domain $\mathcal{D}$ into $\mathcal{H}^\infty = \overline{\mathcal{D}}$ which is the infinitesimal generator of a strongly continuous semi-group of transition operators $T(t)$ for $t \geq 0$. Finally we require that B: $\mathcal{U} \to \mathcal{H}^\infty$ be a bounded linear operator.

The celebrated "Spillover Problem" now has an exact formulation by means of:

THEOREM. (Balakrishnan, [5], p. 233.) If B is compact, then $\mathcal{S}$ is *NOT exactly controllable.*

A compact operator is one which can be approximated arbitrarily closely by an operator whose range is finite-dimensional. Therefore the practical import of the preceding theorem can be phrased as: *if a linear system has infinitely many independent modes of motion, it cannot be controlled completely with a finite-dimensional actuator suite.*

This suggests that complete control of a flexible structure by a finite actuator suite is foredoomed to impossibility. However, there may be a way to circumvent this difficulty. Note that the preceding theorem has been proved only in the case that the dynamical system $\mathcal{S}$ is linear. The purpose of the present paper is to demonstrate that for many flexible structure problems, such nonlinear mechanisms as Balakrishnan-Taylor damping will couple the higher order modes of motion to the lower order modes in such a way that only a finite number of the lower order modes is functionally independent. This suggests that a finite actuator suite could control such a system. However, we defer consideration of the control problem and deal here only with the free motion of an uncontrolled, but intrinsically nonlinear system. Our purpose is to stimulate further research into this approach rather than to present a finished theory.

Liapunov-Schmidt Bifurcation Theory

For the reader's convenience we recall the salient features of this theory from a purely formal point of view. The details of Banach space rigor can be found in pages 173-177 of Deimling [2] and other texts on nonlinear functional analysis [3], [4].

Let $\mathcal{L}$ denote a Fredholm operator, which may be singular, i.e. there may exist elements $u \neq 0$ such that $\mathcal{L}u = 0$. Let $\mathcal{K}$ denote a pseudo-inverse of $\mathcal{L}$, i.e. a linear operator such that

$$\mathcal{K}\mathcal{L}\mathcal{K} = \mathcal{K}\ , \qquad \mathcal{L}\mathcal{K}\mathcal{L} = \mathcal{L}. \tag{2}$$

Now define projection operators

$$\mathcal{P} = \mathcal{I} - \mathcal{K}\mathcal{L}, \qquad Q = \mathcal{I} - \mathcal{L}\mathcal{K}; \tag{3}$$

it is readily verified that $\mathcal{P}^2 = \mathcal{P}$, $Q^2 = Q$, i.e. these operators are idempotent, which justifies referring to them as projection operators. Note that $\mathcal{P}$ is a right zero of $\mathcal{L}$, and Q is a left zero of $\mathcal{L}$.

Let $\mathcal{F} = \mathcal{F}\{\cdot\}$ denote a nonlinear operator. Then it is easy to verify the equivalence of the implicit equation problem

$$(*) \qquad \mathcal{L}x = \mathcal{F}\{x\}, \quad \mathcal{L}\mathcal{P} = 0, \quad Q\mathcal{L} = 0, \tag{4}$$

and the problem

$$(**) \qquad x = \mathcal{P}x + \mathcal{K}\mathcal{F}\{x\}, \quad Q\mathcal{F}\{x\} = 0. \tag{5}$$

Now define $u = \mathcal{P}x$ and verify that $\mathcal{P}u = u$; then we can replace $(**)$ by

$$\text{(definition)} \qquad x = u + v, \tag{6}$$

$$\textbf{(AUXILIARY EQUATION)} \qquad v = \mathcal{K}\mathcal{F}\{u + v\}, \tag{7}$$

$$\textbf{(BIFURCATION EQUATION)} \qquad Q\mathcal{F}\{u + v\} = 0. \tag{8}$$

Another name for the Bifurcation Equations is Branching Equations.

Suppose that the right-hand side of (7), regarded as a function of v, has a global Lipschitz constant less than unity. Then by the well-known principle of geometric convergence of Contraction Mappings we may, for each fixed u, define a nonlinear mapping $v = \mathcal{G}\{u\}$ as

$$v = \lim_{k\to\infty} v^k, \qquad v^{k+1} = \mathcal{K}\mathcal{F}\{u + v^k\}, \quad v^0 = 0, \quad (\ k = 0,\ 1,\ 2,\ 3,\ \cdots\ ). \tag{9}$$

Here $\mathcal{G}$ is the resolvent of the auxiliary equation in the sense that

$$\mathcal{G} \equiv \mathcal{K}\mathcal{F}\{u + \mathcal{G}\}. \tag{10}$$

Hence we may eliminate the auxiliary equation and replace v in the bifurcation equation by $\mathcal{G}$ to obtain a new finite-dimensional equation

$$f(u) \equiv Q\mathcal{F}\{u + \mathcal{G}\{u\}\} = 0, \tag{11}$$

which is equivalent to the original infinite-dimensional implicit equation. Thus

$$(*) \iff (**) \iff (***) \quad x = u + \mathcal{G}\{u\}, \quad \mathcal{L}u = 0, \quad \mathcal{P}\mathcal{G}\{u\} = 0, \quad \mathcal{P}u = u, \quad f(u) = 0. \tag{12}$$

If the original functional equation was analytic, then the final finite-dimensional equation $f(u) = 0$ will also be analytic.

If $\mathcal{L}$ was non-singular, then $\mathcal{K} = \mathcal{L}^{-1}$, whence $\mathcal{P} = 0$, $Q = 0$, $u = 0$, the bifurcation equation does not arise, and the resolvability of the auxiliary equation is equivalent to the resolvability of the original equation:

$$\mathcal{L}x = \mathcal{F}\{x\} \quad \iff \quad x = \mathcal{L}^{-1}\mathcal{F}\{x\} = \mathcal{G}\{0\}. \tag{13}$$

Finally, if $\mathcal{L}$ was singular, then the linear part $F = f_x(0) = (\partial f_i/\partial x_j)$ of $f(x)$ is necessarily also singular. Typically then the solutions of $f(x) = 0$ will not be unique and one studies the branching of these solutions by such methods as Newton's polygon.

Deflection of an Extensible, Nonlinearly Damped Beam

Let u denote the normal deflection from equilibrium. Then the vibrations of the beam can be described by $u = u(t,x)$, $0 \le x \le L$, $0 \le t < +\infty$, which satisfies the PDE

$$\rho \cdot u_{tt} + EI \cdot u_{xxxx} = C \cdot u_{xxt} + \left(H + \frac{EA_c}{L} \cdot \Phi + \Gamma \cdot \Psi \right) u_{xx}, \tag{14a}$$

$$\Phi \equiv (1/2) \cdot \int_0^L (u_x)^2 \cdot dx, \tag{14b}$$

$$\Psi \equiv \left[\int_0^L (u_x u_{xt}) \cdot dx \right]^{2(n+\beta)+1}, \quad (0 \le \beta < 1/2), \quad (n = 0, 1, 2, 3, \cdots), \tag{14c}$$

where

$t \in \mathbb{R}_+ \equiv [0,+\infty) \equiv \{ \tau \mid 0 \le \tau < +\infty \}$,

$x \in J \equiv [0,1] \equiv \{ \xi \mid 0 \le \xi \le 1 \}$,

ρ = density,
E = Young's modulus of elasticity,
I = cross-sectional moment of inertia,
C = coefficient of viscous damping,
H = axial force (tension or compression)
A_c = cross-sectional area,
L = length,
Γ = Balakrishnan-Taylor damping coefficient.

Now define the constants

$$a_0 = H/\rho, \quad a_1 = EA_c/(2\rho L), \quad a_2 = EI/\rho, \quad a_3 = C/\rho, \quad a_4 = \Gamma/\rho \tag{15}$$

Then the PDE (14) can be expressed as

$$u_{tt} + a_2 \cdot u_{xxxx} = a_3 \cdot u_{xxt} + \left(a_0 + a_1 \cdot \int_0^L (u_x)^2 \cdot dx + a_4 \cdot \left[\int_0^L (u_x u_{xt}) \cdot dx \right]^{2(n+\beta)+1} \right) \cdot u_{xx}. \tag{16}$$

Boundary Conditions

As usual, we require that

$$u(t,0) = 0, \qquad u(t,L) = 0, \tag{17a}$$

$$u_{xx}(t,0) = 0, \qquad u_{xx}(t,L) = 0. \tag{17b}$$

Initial Conditions

Let ϕ and ψ be functions of x defined on J with the following smoothness requirements. The function ϕ should be continuously once differentiable on J and its second derivative ϕ'' should exist almost everywhere on J and be [Lebesgue] square-integrable on J. The function ψ should be continuous on J, which of course implies that it is [Riemann] square-integrable on J. These smoothness requirements may be summarized as:

$$\phi \in C^{(1)}(J), \qquad \phi'' \in L_2(J), \tag{17c}$$

$$\psi \in C^{(0)}(J), \quad \Rightarrow \quad \psi \in L_2(J). \tag{17d}$$

Now ϕ and ψ are used to define the initial conditions on u as follows:

$$u(0,x) \equiv \phi(x), \qquad u_t(0,x) \equiv \psi(x), \tag{17e}$$

$$\phi(0) = \phi(L) = 0, \qquad \psi(0) = \psi(L) = 0. \tag{17f}$$

Normalized Constants

For future convenience, we define

$$b_0 = (\pi/L)^2 \cdot a_0, \quad b_1 = (\pi/L)^4 \cdot (L/2) \cdot a_1, \quad b_2 = (\pi/L)^4 \cdot a_2, \tag{18a}$$

$$b_3 = (\pi/L)^2 \cdot a_3, \quad b_4 = (\pi/L)^{4[1+n+\beta]} \cdot (L/2)^{2(n+\beta)+1} \cdot a_4. \tag{18b}$$

Function Space Coordinatization

Define the complete orthonormal set $\{e_k\}$ on $L_2(J)$ by

$$e_k = e_k(x) \equiv \sin(k\pi[x/L]), \qquad (\ k = 1, 2, 3, \cdots\ ). \tag{19}$$

Assumptions (17c,d) imply that there exist (as l.i.m.) sequences $\{\alpha_k\}$, $\{\beta_k\}$ such that

$$\phi = \phi(x) = \sum_{k=1}^{\infty} \alpha_k \cdot e_k(x), \qquad \sum_{k=1}^{\infty} k^4 \cdot \alpha_k^2 < +\infty, \tag{20a}$$

$$\psi = \psi(x) = \sum_{k=1}^{\infty} \beta_k \cdot e_k(x), \qquad \sum_{k=1}^{\infty} \beta_k^2 < +\infty. \tag{20b}$$

Now we can seek to find a sequence of time-varying functions $\{u_k(t)\}$ such that

$$u(t,x) = \sum_{k=1}^{\infty} u_k(t) \cdot e_k(x), \tag{21}$$

where, by (17f), the initial values and initial rates of the $\{u_k(t)\}$ must satisfy

$$u_k(0) = \alpha_k, \qquad \dot{u}_k(0) = \beta_k, \qquad (\ k = 1, 2, 3, \cdots\ ). \tag{22}$$

Infinite System of ODEs

Insert the series expansion (21) into the PDE (16) and use the orthonormality property of the complete basis $\{e_k\}$ to derive an equivalent infinite system of ODEs:

$$\ddot{u}_k + b_3 k^2 \cdot \dot{u}_k + (b_0 + b_1 \hat{\Phi} + b_4 \hat{\Psi} + b_2 k^2) \cdot k^2 \cdot u_k = 0, \qquad (\ k = 1, 2, 3, \cdots\ ), \tag{23a}$$

$$\hat{\Phi} \equiv \sum_{j=1}^{\infty} j^2 \cdot u_j^2\ , \tag{23b}$$

$$\hat{\Psi} \equiv \left[\sum_{j=1}^{\infty} j^2 \cdot u_j \cdot \dot{u}_j \right]^{2(n+\beta)+1}, \tag{23c}$$

where the solutions of (23) are required to satisfy the initial conditions (22). As an alternative to (23a), in which the linear and nonlinear terms are displayed separately, we may write

$$\ddot{u}_k + b_3 k^2 \cdot \dot{u}_k + (b_0 + b_2 k^2) \cdot k^2 \cdot u_k = -\tilde{\Phi} \cdot k^2 \cdot u_k, \qquad (\ k = 1, 2, 3, \cdots\ ), \tag{23d}$$

$$\tilde{\Phi} \equiv b_1 \hat{\Phi} + b_4 \hat{\Psi} \equiv b_1 \cdot \sum_{j=1}^{\infty} j^2 \cdot u_j^2 + b_4 \cdot \left[\sum_{j=1}^{\infty} j^2 \cdot u_j \cdot \dot{u}_j \right]^{2(n+\beta)+1} . \tag{23e}$$

Energy Integral

Multiply each equation of (23a) by $2\dot{u}_k$ and sum over all k, using the identities $2\ddot{u}_k \cdot \dot{u}_k \equiv (\dot{u}_k^2)^{\cdot}$ and $2u_k \cdot \dot{u}_k \equiv (u_k^2)^{\cdot}$, to obtain

$$\sum_{k=1}^{\infty} \dot{u}_k^2 + b_0 \cdot \sum_{k=1}^{\infty} k^2 \cdot u_k^2 + b_2 \cdot \sum_{k=1}^{\infty} k^4 \cdot u_k^2 + (b_1/2) \cdot \left[\sum_{k=1}^{\infty} k^2 \cdot u_k^2 \right]^2 +$$

$$+ 2b_3 \cdot \int_0^t \left[\sum_{k=1}^{\infty} k^2 \cdot \dot{u}_k^2 \right] dt + (b_4/2) \cdot \int_0^t \left[\sum_{k=1}^{\infty} k^2 \cdot u_k \cdot \dot{u}_k \right]^{2(n+\beta)+1} dt \equiv \tag{24a}$$

$$\equiv \mathcal{E}_0 \equiv \textit{constant} \equiv \tag{24b}$$

$$\equiv \sum_{k=1}^{\infty} \beta_k^2 + \sum_{k=1}^{\infty} (b_0 + b_2 k^2) \cdot k^2 \cdot \alpha_k^2 + (b_1/2) \left[\sum_{k=1}^{\infty} k^2 \cdot \alpha_k^2 \right]^2 \equiv \tag{24c}$$

$$\equiv \frac{2}{L} \int_0^L \psi^2 \, dx + b_0 \cdot \frac{2}{L} \int_0^L (\phi')^2 \, dx + b_2 \cdot \frac{2}{L} \int_0^L (\phi'')^2 \, dx + (b_1/2) \cdot \left[\frac{2}{L} \int_0^L (\phi')^2 \, dx \right]^2 \equiv \tag{24d}$$

$$\equiv \bar{\psi}^2 + b_0 \cdot (\bar{\phi}')^2 + b_2 \cdot (\bar{\phi}'')^2 + (b_1/2) \cdot \left[(\bar{\phi}')^2 \right]^2 , \tag{24e}$$

where in the last expression we have used an obvious notation for the mean-square values of ψ, ϕ', and ϕ'' on $J = [0,L]$.

A key technique in what follows is the use of (24) to obtain *a priori* bounds on the solutions of (23).

Vector Notation

For convenience we shall denote the infinite column whose rows are ku_k by $\mathbf{x}$, and we shall partition $\mathbf{x}$ into a finite-dimensional component $\mathbf{u}$ and an infinite-dimensional component $\mathbf{v}$ as follows:

$$\mathbf{x} = \begin{pmatrix} \mathbf{u} \\ \mathbf{v} \end{pmatrix} \equiv (ku_k), \qquad \mathbf{u} \in \mathbb{R}^N, \qquad \mathbf{v} \in \mathbb{R}^{\infty}, \tag{25a}$$

$$\mathbf{u}_k = ku_k, \quad \mathbf{v}_k = kv_k, \quad (v_k \equiv u_k , \; k \geq N). \tag{25b}$$

A Priori Bounds

Obviously

$$\|\mathbf{x}\|^2 \equiv \|\mathbf{u}\|^2 + \|\mathbf{v}\|^2 \equiv \sum_{k=1}^{\infty} k^2 \cdot u_k^2 , \tag{26}$$

while from (24a) it is easy to infer that

$$(b_0 + b_2) \cdot \|\mathbf{x}\|^2 + (b_1/2) \cdot \|\mathbf{x}\|^4 \leq \mathcal{E}_0 , \tag{27}$$

which implies that

$$\|\mathbf{x}\| \leq R, \tag{28}$$

where (choosing the numerically stable quadratic root formula)

$$R \equiv (2\mathcal{E}_0/\{(b_0 + b_2) + [(b_0 + b_2)^2 + 2b_1\mathcal{E}_0]^{1/2}\})^{1/2} \leq (\mathcal{E}_0/[b_0 + b_2])^{1/2}. \tag{29}$$

Similarly,

$$(N+1)^2 \cdot \|\mathbf{v}\|^2 \equiv (N+1)^2 \cdot \sum_{k=N+1}^{\infty} k^2 \cdot v_k^2 \leq \sum_{k=N+1}^{\infty} k^4 \cdot v_k^2 \equiv \sum_{k=N+1}^{\infty} k^4 \cdot u_k^2 \leq \sum_{k=1}^{\infty} k^4 \cdot u_k^2 \leq \mathcal{E}_0/b_2 , \tag{30}$$

and

$$\sum_{k=1}^{\infty} \dot{u}_k^2 \leq \mathcal{E}_0 , \tag{31}$$

which together with (30) yields, via the Cauchy-Schwarz inequality,

$$\sum_{j=1}^{\infty} j^2 \cdot u_j \cdot \dot{u}_j \leq \left\{ \left(\sum_{k=1}^{\infty} \dot{u}_k^2 \right) \cdot \left(\sum_{k=1}^{\infty} k^4 \cdot u_k^2 \right) \right\}^{1/2} \leq (\mathcal{E}_0[\mathcal{E}_0/b_2])^{1/2} = \mathcal{E}_0/(b_2)^{1/2}. \tag{32}$$

Finally, using (29) and (32),

$$\tilde{\Phi} \equiv b_1 \cdot \|\mathbf{x}\|^2 + b_4 \cdot (\mathbf{x} \cdot \dot{\mathbf{x}})^{2(n+\beta)+1} \leq b_1 \cdot R^2 + b_4 \cdot [\mathcal{E}_0/(b_2)^{1/2}]^{2(n+\beta)+1} \leq b_5 \cdot \mathcal{E}_0 , \tag{33}$$

$$b_5 \equiv [b_1/(b_0 + b_2)] + b_4/(b_2)^{n+\beta+(1/2)}, \tag{34}$$

where we have used the assumption that

$$\mathcal{E}_0 < 1. \tag{35}$$

Now let $\mathbb{B}^N(\rho)$ and $\mathbb{B}^\infty(\rho)$ denote balls of radius ρ in $\mathbb{R}^N$ and $\mathbb{R}^\infty$ where $\mathbf{x} = \mathbf{u} \otimes \mathbf{v}$ is considered to be an element of the Cartesian product $\mathbb{R}^N \otimes \mathbb{R}^\infty$; we shall show that for all sufficiently large values of N, $\mathbb{B}^N(R_\varepsilon) \otimes \mathbb{B}^\infty(R_N)$ is a subset of the ball defined in $\mathbb{R}^N \otimes \mathbb{R}^\infty$ by (28), where $0 < \varepsilon < 1$ is an arbitrarily small positive constant, where

$$R_\varepsilon \equiv R \cdot (1 - \varepsilon)^{1/2}, \tag{36}$$

$$R_N \equiv (\mathcal{E}_0/b_2)^{1/2}/(N + 1), \tag{37}$$

and where

$$N + 1 \geq (\mathcal{E}_0/b_2)^{1/2}/(R\varepsilon^{1/2}). \tag{38}$$

In fact,

$$\|\mathbf{x}\|^2 \equiv \|\mathbf{u}\|^2 + \|\mathbf{v}\|^2 \leq (R_\varepsilon)^2 + (R_N)^2 \leq R^2 \cdot (1 - \varepsilon) + R^2 \cdot \varepsilon \equiv R^2. \tag{39}$$

Initial Conditions

Define

$$\mathbf{a} \equiv (k \cdot \alpha_k) \in \mathbb{R}^N, \quad \mathbf{a}^{(N)} \equiv ([N + k] \cdot \alpha_{N+k}) \in \mathbb{R}^\infty, \tag{40a}$$

and note that

$$\mathbf{a} = \mathbf{u}(0), \quad \mathbf{a}^{(N)} = \mathbf{v}(0). \tag{40b}$$

Next, define

$$\mathbf{b} \equiv (\beta_k) \in \mathbb{R}^N, \quad \mathbf{b}^{(N)} \equiv (\beta_{N+k}) \in \mathbb{R}^\infty, \tag{40c}$$

and note that

$$\mathbf{b} = \mathcal{D}^{-1}\cdot\dot{\mathbf{u}}(0), \qquad \mathbf{b}^{(N)} = \mathcal{D}_N^{-1}\cdot\dot{\mathbf{v}}(0). \tag{40d}$$

where the bounded linear operators $\mathcal{D}^{-1}$ and $\mathcal{D}_N^{-1}$ are represented by matrices

$$\mathcal{D}^{-1} = diag\ (\ 1/k\ ), \qquad \mathcal{D}_N^{-1} = diag\ (\ 1/[k+N]\ ). \tag{40e}$$

Resolution of Linear Part

In component form, the infinite ODE can be written, for ($k = 1, 2, 3, \cdots$), as

$$\ddot{u}_k + b_3k^2\cdot\dot{u}_k + (b_0+b_2k^2)\cdot k^2\cdot u_k = -\left\{b_1\sum_{j=1}^{\infty} j^2\cdot u_j^2 + b_4\left[\sum_{j=1}^{\infty} j^2\cdot u_j\cdot\dot{u}_j\right]^{2(n+\beta)+1}\right\}k^2\cdot u_k. \tag{41}$$

Alternatively, in vector notation

$$\mathcal{D}_0^{-1}\ddot{\mathbf{x}} + b_3\cdot\mathcal{D}_0\dot{\mathbf{x}} + (b_0 + b_2\cdot\mathcal{D}_0^2)\cdot\mathcal{D}_0\mathbf{x} = -\{\ b_1\cdot\|\mathbf{x}\|^2 + b_4\cdot(\mathbf{x}\cdot\dot{\mathbf{x}})^{2(n+\beta)+1}\ \}\cdot\mathcal{D}_0\mathbf{x}, \tag{42}$$

b_0 = axial force coefficient, (43a)

b_1 = extensibility coefficient, (43b)

b_2 = elasticity coefficient, (43c)

b_3 = viscous damping coefficient, (43d)

b_4 = Balakrishnan-Taylor nonlinear damping coefficient, (43e)

where the unbounded linear operator $\mathcal{D}_0$ is defined by the infinite matrix obtained by taking $N = 0$ in the reciprocal of (40e). Upon multiplying (42) through by $\mathcal{D}_0^{-1}\dot{\mathbf{x}}$ and integrating, we obtain the energy integral (24) in the form

$$\|\mathcal{D}_0^{-1}\dot{\mathbf{x}}\|^2 + b_0\cdot\|\mathbf{x}\|^2 + (b_1/2)\cdot\|\mathbf{x}\|^4 + b_2\cdot\|\mathcal{D}_0\mathbf{x}\|^2 + 2\int_0^t\{b_3\cdot\|\dot{\mathbf{x}}\|^2 + b_4\cdot(\mathbf{x}\cdot\dot{\mathbf{x}})^{2(n+\beta+1)}\}\,dt \equiv$$

$$\equiv \mathcal{E}_0 \equiv \|\mathbf{b}^{\infty}\|^2 + b_0\cdot\|\mathbf{a}^{\infty}\|^2 + (b_1/2)\cdot\|\mathbf{a}^{\infty}\|^4 + b_2\cdot\|\mathcal{D}_0\mathbf{a}^{\infty}\|^2, \tag{44a}$$

where as before

$$\mathbf{a}^{\infty} = \mathbf{a}\otimes\mathbf{a}^N, \qquad \mathbf{b}^{\infty} = \mathbf{b}\otimes\mathbf{b}^N. \tag{44b}$$

For future use define ε_N and δ_N by

$$(N+1)^2\cdot\|\mathbf{a}^N\|^2 \le \sum_{k=N+1}^{\infty} k^4\cdot\alpha_k^2 \equiv \varepsilon_N^2 \to 0, \qquad N \to +\infty, \tag{45a}$$

$$\|\mathbf{b}^N\|^2 = \sum_{k=N+1}^{\infty}\beta_k^2 \equiv \delta_N^2 \to 0, \qquad N \to +\infty, \tag{45b}$$

Homogeneous Linear Part

Henceforth we shall assume that the system is underdamped, i.e. that

$$b_3 < 2(b_2)^{1/2}. \tag{46}$$

The characteristic polynomial of the homogeneous part of (41) is

$$\lambda_k^2 + b_3 k^2 \cdot \lambda_k + (b_0 + b_2 k^2) \cdot k^2 = 0, \tag{47a}$$

which has roots

$$\lambda_k = -\mu_k \pm i \cdot \nu_k , \quad (i^2 = -1), \tag{47b}$$

$$\mu_k = k^2 \cdot \mu_0 , \quad \nu_k = k^2 \cdot \omega_k , \tag{47c}$$

$$\mu_0 \equiv b_3/2 , \quad \omega_k \equiv (1/2) \cdot \{4b_2 - (b_3)^2 + 4(b_0/k^2)\}^{1/2} , \tag{47d}$$

where, obviously,

$$\mu_0/\omega_k \leq b_3/\{4b_2 - (b_3)^2\}^{1/2}. \tag{47e}$$

From (47a) the general solution of the homogeneous linear part of (41) is of the form

$$u_k = exp(-\mu_k t) \cdot \{A_k \cdot cos(\nu_k t) + B_k \cdot sin(\nu_k t)\}, \tag{47f}$$

where A_k and B_k are arbitrary constants. Specializing these constants in order to use Lagrange's variation of constants formula to re-express (41) in terms of an impulse response convolution with the right-hand side we get

$$u_k(t) = d_k^{(1)}(t) \cdot \alpha_k + (d_k^{(2)}(t)/k) \cdot \beta_k - \int_0^t d_k^{(3)}(t - \tau) \, \Phi \, u_k(\tau) \, d\tau, \tag{47g}$$

where

$$d_k^{(1)}(t) \equiv exp(-\mu_k t) \cdot \{cos(\nu_k t) + (\mu_0/\omega_k) \cdot sin(\nu_k t)\}, \tag{48a}$$

$$d_k^{(2)}(t) \equiv exp(-\mu_k t) \cdot \{[1/(k \cdot \omega_k)] \cdot sin(\nu_k t)\}, \tag{48b}$$

$$d_k^{(3)}(t) \equiv k \cdot d_k^{(2)}(t). \tag{48c}$$

Next, define three finite and three infinite diagonal matrices as

$$D_j \equiv diag(d_k^{(j)}) : \mathbb{R}^N \rightarrow \mathbb{R}^N , \quad (j = 1, 2, 3), \quad (k = 1, 2, 3, \cdots, N), \tag{49a}$$

$$D_{J,N} \equiv diag(d_{k+N}^{(j)}) : \mathbb{R}^\infty \rightarrow \mathbb{R}^\infty , \quad (j = 1, 2, 3), \quad (k = 1, 2, 3, \cdots). \tag{49b}$$

Finally, multiply (47g) through by k in order to convert to vector notation:

$$\mathbf{x}(t) = D_{1,0}(t) \cdot \mathbf{a}^\infty + D_{2,0}(t) \cdot \mathbf{b}^\infty - \int_0^t D_{3,0}(t - \tau) \cdot \tilde{\Phi} \cdot \mathbf{x}(\tau) \, d\tau, \tag{50a}$$

$$\dot{\mathbf{x}}(t) = \dot{D}_{1,0}(t) \cdot \mathbf{a}^\infty + \dot{D}_{2,0}(t) \cdot \mathbf{b}^\infty - \int_0^t \dot{D}_{3,0}(t - \tau) \cdot \tilde{\Phi} \cdot \mathbf{x}(\tau) \, d\tau, \tag{50b}$$

$$\tilde{\Phi} \equiv b_1 \cdot \|\mathbf{x}\|^2 + b_4 \cdot (\mathbf{x} \cdot \dot{\mathbf{x}})^{2(n+\beta)+1}. \tag{50c}$$

(In deriving (50b) we used the fact that $D_{3,0}(0) \equiv 0$.) The fact that we have been able to reformulate the original PDE boundary-value and initial-value problem in the form (50) is the equivalent of (13), i.e the non-singular case, wherein there is no requirement for a Bifurcation Equation since the entire system now has the form of the Auxiliary Equation. If now we can prove that the iteration, for $m = 0, 1, 2, 3, \cdots$,

$$\mathbf{x}^{m+1}(t) \;=\; D_{1,0}(t)\cdot\mathbf{a}^{\infty} + D_{2,0}(t)\cdot\mathbf{b}^{\infty} - \int_0^t D_{3,0}(t-\tau)\cdot\tilde{\Phi}(\mathbf{x}^m,\dot{\mathbf{x}}^m)\cdot\mathbf{x}^m(\tau)\,d\tau, \tag{51a}$$

$$\mathbf{x}^0(t) \;\equiv\; 0, \qquad \mathbf{x}^1(t) \;\equiv\; D_{1,0}(t)\cdot\mathbf{a}^{\infty} + D_{2,0}(t)\cdot\mathbf{b}^{\infty}, \tag{51b}$$

converges, then we have constructed a solution of the original problem.

To avoid certain difficulties, we shall consider this iterative solution only for the problem

$$b_4 = 0, \qquad \tilde{\Phi} \;=\; \tilde{\Phi}(\mathbf{x}) \;\equiv\; b_1\cdot\|\mathbf{x}\|^2; \tag{52}$$

the more general problem will be approached by a non-constructive homotopy method. Consider now the fixed-point problem

$$\mathbf{x} \;=\; \underset{(t>0)}{\mathcal{F}}\{\mathbf{x}\}, \qquad \|\mathbf{x}\|_{\infty} \;\equiv\; sup\;\; \|\mathbf{x}(t)\|, \tag{53a}$$

$$\mathcal{F} \;\equiv\; \mathcal{F}\{\mathbf{x}(\cdot)\}(t) \;\equiv\; D_{1,0}(t)\cdot\mathbf{a}^{\infty} + D_{2,0}(t)\cdot\mathbf{b}^{\infty} - \int_0^t D_{3,0}(t-\tau)\cdot\tilde{\Phi}(\mathbf{x}(\tau))\cdot\mathbf{x}(\tau)\,d\tau. \tag{53b}$$

We want to find a Lipschitz constant for $\mathcal{F}$, i.e. a constant κ such that

$$\|\mathcal{F}\{\mathbf{x}^{j+1}\} - \mathcal{F}\{\mathbf{x}^j\}\|_{\infty} \;\le\; \kappa\cdot\|\mathbf{x}^{j+1} - \mathbf{x}^j\|_{\infty} \qquad \forall\;\; \mathbf{x}^j \in \mathbb{B}^{\infty}(R), \quad (\; j = 0,\, 1,\, 2,\, \cdots\;). \tag{54}$$

Later we shall prove that

$$\int_0^{+\infty} \|D_{3,0}(\tau)\|\,d\tau \;\le\; b_6 \;\equiv\; 4/\{b_3[4b_2 - (b_3)^2]^{1/2}\}. \tag{55}$$

In the Hilbert space norm of $\mathbb{R}^{\infty}$ it is clear that

$$\|\,\{\|\mathbf{x}^2\|^2\}\,\mathbf{x}^2 - \{\|\mathbf{x}^1\|^2\}\,\mathbf{x}^1\,\| \;\equiv\; \|\,\{\|\mathbf{x}^2\|^2\}\,(\mathbf{x}^2 - \mathbf{x}^1) + \{(\mathbf{x}^2 + \mathbf{x}^1)\cdot(\mathbf{x}^2 - \mathbf{x}^1)\}\,\mathbf{x}^1\,\| \;\le$$

$$\le\; \{\|\mathbf{x}^1\|^2 + \|\mathbf{x}^2\|^2 + \|\mathbf{x}^1\|\cdot\|\mathbf{x}^2\|\}\cdot\|\mathbf{x}^2 - \mathbf{x}^1\| \;\le 3R^2\cdot\|\mathbf{x}^2 - \mathbf{x}^1\|. \tag{56}$$

Hence we may take

$$\kappa \;=\; 3R^2 b_1\cdot b_6 \;\le\; 3\{b_1\cdot b_6/[b_0 + b_2]\}\mathcal{E}_0 \;\le\; \vartheta \;<\; 1, \tag{57a}$$

if $\mathcal{E}_0$ is taken to be sufficiently small that

$$\mathcal{E}_0 \;<\; (1/12)\cdot\vartheta\cdot\{(b_0 + b_2)\cdot b_3\cdot[4b_2 - (b_3)^2]^{1/2}\}/b_1. \tag{57b}$$

The *a priori* bounds (28)-(29) apply to the first iterate (51b) and so the first iterate is inside the ball $\mathbb{B}^{\infty}(R)$; now, using (54) with $j = 0,\, 1,\, 2,$ etc. it is clear that $\mathbf{x}^2$, $\mathbf{x}^3$, all remain in the ball provided that $\|\mathbf{x}^1\|_{\infty} + \vartheta\cdot\|\mathbf{x}^1\|_{\infty} + \vartheta^2\cdot\|\mathbf{x}^1\|_{\infty} + \cdots \;< R$, i.e. summing the geometric series, provided that

$$\{\|\mathbf{x}^1\|_{\infty}/(1 - \vartheta)\} \;<\; R, \tag{58a}$$

which is readily obtainable simply by taking ϑ so small that

$$\vartheta \;<\; 1 - (\|\mathbf{x}^1\|_{\infty}/R). \tag{58b}$$

Now we can apply the well known principle of contraction mappings (also called the Banach fixed point theorem and Caccioppoli's fixed point theorem) to prove that the map (53) has a unique fixed point in the ball $\mathbb{B}^{\infty}(R)$ and that this fixed point can be computed constructively by the iterative procedure just described.

Next, replace the coefficient b_4 by μb_4, where $0 \le \mu \le 1$, and note that the *a priori* bounds previously derived remain valid for all values of $\mu \in [0,1]$. Hence we may infer the existence of a solution for all $\mu \in [0,1]$ by a homotopy method. For sufficiently small values of μ, this is Poincaré's method of analytic continuation of solutions of functional equations. Here the fact that the problem was non-singular at $\mu = 0$ implies that the Leray-Schauder Index [2], [3] or topological degree of the map is of magnitude unity at $\mu = 0$. Consequently the existence of the *a priori* bound for any solution on $0 \le \mu \le 1$ implies that there is a continuum of solutions connecting the solution at $\mu = 0$ with one at $\mu = 1$. Thus in summary we have proved the following result.

Existence Theorem

THEOREM. If $\bar{\phi}'$, $\bar{\phi}''$, and $\bar{\psi}$ [as defined in (24d-e)] are all sufficiently small, then the nonlinear functional PDE (16) has at least one solution which exists for all $t \ge 0$ and satisfies the boundary conditions (17a-b) and initial conditions (17e) as well as the *a priori* bounds

$$\frac{2}{L}\int_0^L [u_x(t,x)]^2\, dx \;\le\; R^2 \;\le\; \mathcal{E}_0/(b_0 + b_2), \tag{59}$$

$$\int_0^{+\infty}\left\{\frac{2}{L}\int_0^L [u_{xt}(t,x)]^2\, dx\right\} dt \;\le\; \mathcal{E}_0/(2b_3), \tag{60}$$

where

$$\mathcal{E}_0 \;\equiv\; \bar{\psi}^2 + b_0\cdot(\bar{\phi}')^2 + b_2\cdot(\bar{\phi}'')^2 + (b_1/2)\cdot\left[(\bar{\phi}')^2\right]^2. \tag{62}$$

Proof. Existence has already been proved. Equation (59) holds because the left-hand side of (59) is by (26) equal to $\|\mathbf{x}\|^2$ which in (28) is proved smaller than R^2. Equation (60) follows from inspection of (24a) and (24b), which hold for all $t \ge 0$, making it permissible to let $t \to +\infty$.

Rigorous Truncation

In "naive truncation" one simply sets $\mathbf{v} \equiv 0$, i.e. in (41) one takes

$$u_k(t) \;\equiv\; 0, \quad (\, k = N+1,\ N+2,\ N+3,\ \cdots\,). \tag{63}$$

Here we shall prove that the solution proved above to exist can be derived as in the Liapunov-Schmidt bifurcation theory, and that, for all $t \ge 0$,

$$\|\mathbf{v}(t)\| \;\le\; R_N\cdot exp(-b_3[N+1]^2\cdot t/2) \;\le\; R\varepsilon^{1/2}, \tag{64}$$

provided that N is taken larger than the lower bounds in (38) and in (68b)-(69) below.

By inspection of (50), we can express the problem as follows.

Bifurcation Equation

$$\mathbf{u}(t) \;=\; D_1(t)\cdot\mathbf{a} + D_2(t)\cdot\mathbf{b} - \int_0^t D_3(t-\tau)\cdot\tilde{\tilde{\Phi}}\cdot\mathbf{u}(\tau)\, d\tau, \tag{65a}$$

$$\tilde{\tilde{\Phi}} \equiv b_1 \cdot \{\|u\|^2 + \|v\|^2\} + b_4 \cdot (u \cdot \dot{u} + v \cdot \dot{v})^{2(n+\beta)+1}. \tag{65b}$$

Auxiliary Equation

$$v(t) = D_{1,N}(t) \cdot a^N + D_{2,N}(t) \cdot b^N - \int_0^t D_{3,N}(t - \tau) \cdot \tilde{\tilde{\Phi}} \cdot v(\tau)\, d\tau. \tag{65c}$$

More A Priori Bounds

From (48a),

$$|d_k^{(1)}| \leq \{1 + (\mu_0/\omega_k)^2\}^{1/2} \cdot \exp(-\mu_0 k^2 t) \leq \{1 + b_3^2/[4b_2 - (b_3)^2]\}^{1/2} \cdot \exp(-b_3 t/2) \equiv$$

$$\equiv [1/\{1 - (b_3^2/4b_2)\}^{1/2}] \cdot \exp(-b_3 t/2). \tag{66a}$$

The induced Euclidean norm of any diagonal matrix is equal to the absolute value of its (absolutely) largest element. Hence

$$\|D_1\| \leq [1/\{1 - (b_3^2/4b_2)\}^{1/2}] \cdot \exp(-b_3 t/2). \tag{66b}$$

Similarly,

$$\|D_2\| \leq \max_{1 \leq k \leq N} |d_k^{(2)}| \leq \max_{1 \leq k \leq N} \{1/(k\omega_k)\} \cdot \exp(-b_3 t/2) \leq$$

$$\leq [2/\{4b_2 - (b_3^2)\}^{1/2}] \cdot \exp(-b_3 t/2), \tag{66c}$$

$$\|D_3\| \leq \max_{1 \leq k \leq N} \{1/(k\omega_k)\} \cdot \exp(-b_3 t/2) \leq [2/\{4b_2 - (b_3^2)\}^{1/2}] \cdot \exp(-b_3 t/2), \tag{66d}$$

$$\|D_{1,N}\| \leq \{1/[1 - (b_3^2/4b_2)]^{1/2}\} \cdot \exp(-b_3 [N + 1]^2 t/2), \tag{66e}$$

$$\|D_{2,N}\| \leq \{[2/\{4b_2 - (b_3^2)\}^{1/2}]/(N + 1)\} \cdot \exp(-b_3 [N + 1]^2 t/2). \tag{66f}$$

$$\|D_{3,N}\| \leq [2/\{4b_2 - (b_3^2)\}^{1/2}] \cdot \exp(-b_3 [N + 1]^2 t/2). \tag{66g}$$

Integrating (66d), one obtains

$$\int_0^{+\infty} \|D_3(\tau)\|\, d\tau \leq b_6 \equiv 4/\{b_3[4b_2 - (b_3)^2]^{1/2}\}. \tag{66h}$$

Similarly,

$$\int_0^{+\infty} \|D_{3,N}(\tau)\|\, d\tau \leq b_6/(N + 1)^2, \tag{66i}$$

which we have used above in (55) in the case $N = 0$.

Next, from (65c), (66e,f,g), and (45a,b),

$$\|v(t)\| \leq \gamma_N \cdot \exp(-b_3 [N + 1]^2 t/2) + \{b_5 b_6 \mathcal{E}_0/(N + 1)^2\} \cdot \|v(t)\|, \tag{67a}$$

where

$$\gamma_N \equiv \{1/[1 - (b_3^2/4b_2)]^{1/2}\}\|a^N\| + \{[2/\{4b_2 - (b_3^2)\}^{1/2}]/(N + 1)\}\|b^N\| \leq$$

$$\leq \left(\{1/[1 - (b_3^2/4b_2)]^{1/2}\} \cdot \varepsilon_N + [2/\{4b_2 - (b_3^2)\}^{1/2}] \cdot \delta_N \right) / (N+1) \equiv \sigma_N/(N+1). \quad (67b)$$

Consequently, if $\mathcal{E}_0$ is so small that

$$b_6 b_5 \mathcal{E}_0 < 1/2, \quad (68a)$$

i.e. so small that

$$\mathcal{E}_0 < 1/\left(\{8/\{b_3[4b_2 - (b_3)^2]^{1/2}\} \cdot \{[b_1/(b_0 + b_2)] + b_4/(b_2)^{n+\beta+(1/2)}\} \right), \quad (68b)$$

then we may subtract the second term on the right hand side of (67a) from the left hand side, and then, by (45a,b), making N still larger (if necessary) so that

$$\sigma_N/\{1 - (1/2)]\} \equiv 2\sigma_N \leq (\mathcal{E}_0/b_2)^{1/2} \equiv (N+1) \cdot R_N \quad (69)$$

we can ensure that, for all $t \geq 0$,

$$\|\mathbf{v}(t)\| \leq R_N \cdot exp(-b_3[N+1]^2 \cdot t/2) \leq R_N \leq R \cdot \varepsilon^{1/2}, \quad (70)$$

as claimed in (64).

Quite similarly, from (65a) and (68a),

$$\|\mathbf{u}\| \leq \|D_1\| \cdot \|\mathbf{a}\| + \|D_2\| \cdot \|\mathbf{b}\| + (b_6 b_5 \mathcal{E}_0) \cdot \|\mathbf{u}\| \leq \|D_1\| \cdot \|\mathbf{a}\| + \|D_2\| \cdot \|\mathbf{b}\| + (1/2) \cdot \|\mathbf{u}\|, \quad (71a)$$

whence, precisely as before,

$$\{1 - (1/2)\} \cdot \|\mathbf{u}\| \leq \|D_1\| \cdot \|\mathbf{a}\| + \|D_2\| \cdot \|\mathbf{b}\|, \quad (71b)$$

i.e.

$$\|\mathbf{u}\| \leq 2 \cdot \{\|D_1\| \cdot \|\mathbf{a}\| + \|D_2\| \cdot \|\mathbf{b}\|\} \leq R_\varepsilon, \quad (71c)$$

provided only that, by (66b,c)

$$\|\mathbf{a}\| \leq (1/4) \cdot \{1 - (b_3^2/4b_2)\}^{1/2} \cdot R_\varepsilon, \quad (72a)$$

$$\|\mathbf{b}\| \leq (1/8) \cdot \{4b_2 - (b_3^2)\}^{1/2} \cdot R_\varepsilon, \quad (72a)$$

which can be assured simply by taking $\bar{\phi}'$ and $\bar{\psi}$ small enough.

Recalling that, by (62), $\mathcal{E}_0$ can be made arbitrarily small by making the initial mean-square spatial rates of change $\bar{\phi}'$, $\bar{\phi}''$, and $\bar{\psi}$ of u_x, u_{xx} and u_t arbitrarily small, we may conclude that when those initial rates are sufficiently small then every solution of (65a,b) must, for all $t \geq 0$, satisfy *a priori*

$$\|\mathbf{u}(t)\| \leq R_\varepsilon \equiv R \cdot (1 - \varepsilon)^{1/2}, \quad (73a)$$

and every solution of (65b,c) must, for all $t \geq 0$, satisfy *a priori*

$$\|\mathbf{v}(t)\| \leq R_N \leq R \cdot \varepsilon^{1/2}, \quad (73b)$$

whence finally, as in (39), every solution of the combined systems (65a,b,c) must for all $t \geq 0$, satisfy *a priori*

$$\|\mathbf{x}\| \equiv \{\|\mathbf{u}(t)\|^2 + \|\mathbf{v}(t)\|^2\}^{1/2} \leq R. \quad (73c)$$

Conclusion: Naive vs Rigorous Truncation

Let $\mathbf{u}(t)$, $\dot{\mathbf{u}}(t)$ be arbitrarily given continuous N-vector functions and insert them into the nonlinear function $\tilde{\tilde{\Phi}}$ defined in (65b), and then insert this functional of $\mathbf{v}$

only into (65c), giving an infinite system of integral equations for $\mathbf{v}(t)$. As in (28) through (35), the nonlinear term can be made arbitrarily small by taking $\mathcal{E}_0$ arbitrarily small. Similarly the nonlinear term can be made to have a Lipschitz constant less than unity by restrictions upon $\mathcal{E}_0$ as in (53)-(58).

This infinite system of Auxiliary Equations can be solved by one of the methods illustrated above (iteration or homotopy), and the result inserted into the Bifurcation Equations (65a) to provide a *finite-dimensional* system of functional integral equations *exactly-equivalent* to the the original infinite-dimensional system.

When the arbitrarily given $\mathbf{u}(t)$, $\dot{\mathbf{u}}(t)$ in the Auxiliary Equations (65b,c) are taken to be the projection into $\mathbb{R}^N$ of the solution $\mathbf{x}(t) \in \mathbb{R}^N \otimes \mathbb{R}^\infty$ proved to exist in the Theorem concerning (59)-(62), then the Auxiliary Equations have a solution corresponding to the projection $\mathbf{v}(t)$ of $\mathbf{x}(t)$ into $\mathbb{R}^\infty$. The resulting Bifurcation Equations then must be satisfied by the same $\mathbf{u}(t)$ used to define the Auxiliary Equations. However, all of the *a priori* bounds proved above to apply to the solution $\mathbf{x}$ of the complete problem now apply to the projections $\mathbf{u}$, $\mathbf{v}$ of the rigorously truncated problem.

Consequently we can compare the rigorous version of the Bifurcation Equations, namely (65a), with the naively truncated version wherein one sets $\mathbf{v} \equiv 0$, and note that that, as $\mathcal{E}_0$ becomes sufficiently small for the bounds to apply, then as N becomes arbitrarily large the difference between the solutions of the naively truncated version of (65a) and its rigorously truncated version becomes arbitrarily small.

Consequently in attempting to solve the given nonlinear function PDE boundary-value initial-value problem, we may be confident that if we truncate naively for some finite N, the results become arbitrarily accurate as N increases without limit provided that the initial mean-square spatial rates of change $\overline{\phi}'$, $\overline{\phi}''$, and $\overline{\psi}$ of u_x, u_{xx} and u_t are kept sufficiently small.

Further research is needed in order to ascertain whether or not this conclusion would still apply if the homogeneous boundary conditions were replaced by inhomogeneous boundary conditions corresponding to a finite number of actuators.

REFERENCES

Apology: We have mislaid a reprint which treats the above problem in the case wherein $b_4 \equiv 0$ by the present approach in a manner equivalent to (23a) or (41). Although we believe that in this case some of the details above may be novel, this paper should be our **primary reference** and we feel abjectly apologetic to the author and to the journal in question and would appreciate being reminded of this reference by any reader who recognizes it in order to rectify this injustice in a future paper.

Acknowledgements: We are deeply grateful to Dr. A.V. Balakrishnan and to Dr. L. W. Taylor, Jr. for having brought reference [1] to our attention. Also R.W. Bass thanks NASA-Ames Dryden Flight Research Center's support of the UCLA Flight Systems Research Lab (EE Dept.), where an earlier version of this research was done by him.

[1] Balakrishnan, A.V. and L.W. Taylor, Jr. "Distributed Parameter Nonlinear Damping Models for Flight Structures," *Proceedings, "Damping '89"*, Flight Dynamics Lab, Air Force Wright Aeronautical Labs, WPAFB, Feb. 1989.

[2] Deimling, K. *Nonlinear Functional Analysis*, Springer-Verlag, 1985.

[3] Zeidler, E. *Nonlinear Functional Analysis & Appl'ns.*, Springer-Verlag, 1985.

[4] Vainberg, M.M. & V.A. Trenogin. *Theory of Branching of Solutions of Non-linear Functional Equations*, Noordhoff, 1974.

[5] Balakrishnan, A.V. *Functional Analysis*, Springer-Verlag, 1981.

STABILIZATION OF LARGE SPACE STRUCTURES BY LINEAR RELUCTANCE ACTUATORS

N91-22309

SAROJ K. BISWAS HENRY M. SENDAULA

Department of Electrical Engineering
Temple University
Philadelphia, PA 19122.

Abstract

In this paper we consider application magnetic forces for stabilization of vibrations of flexible space structures. We investigate three electromagnetic phenomena, such as, a) magnetic body-force, b) reluctance torque, and c) magnetostriction, and analyse their application for stabilization of a beam. The magnetic body-force actuator utilizes the force that exists between poles of magnets. The reluctance actuator is configured in such a way that the reluctance of the magnetic circuit will be minimum when the beam is straight. Any bending of the beam increases the reluctance and hence generates a restoring torque that reduces bending. The gain of the actuator is controlled by varying the magnetizing current. Since the energy density of a magnetic device is much higher compared to piezo-electric or thermal actuators, it is expected that the reluctance actuator will be more effective in controlling the structural vibrations.

I. INTRODUCTION

The problems of modeling and control of flexible space structures have been a subject of considerable research interest in recent years. These future space vehicles will be large structures consisting of a rigid body and several flexible appendages, such as long beams, solar panels, large antennas etc. It is known that these space structures will possess low structural rigidity, high modal density and low damping. Consequently, in order for them to perform properly some active means of increasing the damping or the energy dissipation must be provided. There is a very large collection of research results available in the literature on the control and stabilization of flexible space structures. The references listed in this paper are only a small cross section of these results, and are not meant to be exhaustive.

Dynamic analysis and control system design of flexible structures are based on two different approaches: a) finite dimensional, and b) infinite dimensional. Although the finite dimensional approach [1 – 7] have been widely investigated in the past, the main objections are modal truncation, lack of *a priori* information of required mode numbers, and control spillover [15]. Because of these reasons, the infinite dimensional approach using partial differential equations appears to be more appropriate. Since large space structures are actually partly rigid and partly flexible, the complete mathematical model requires a combination of both ordinary differential equations and hyperbolic partial differential equations [8 - 16]. Stabilization of flexible space structures through active velocity feedback have been discussed in [8,9,13,14]. A more rigorous analysis of stabilization using semigroup theory is considered in [10,11]. Reference [16] describes the synthesis of optimal controls for this class of systems. Stabilization of flexible systems using thermal [17,18,19] and peizo-electric [20,21] actuators have been investigated in recent years. It has been shown both analytically and experimentally that thermo-elastic damping can be induced in materials by suitable application of thermal gradients. In [20,21] it has been shown that spatially distributed control actuators can be designed using piezoelectric polymers, and that feedback of beam tip angular velocity can be used for stabilization of vibrations of a beam.

In this paper, we investigate application of magnetic forces for stabilization of elastic structures. Magnetic forces and torques are developed in ferromagnetic systems in a variety of ways. Here we discuss three electromagnetic phenomena which have very good potential of stabilizing a vibrating structure; these are: a) magnetic body-force, b) reluctance torque, and c) magnetostriction. Magnetic body-force actuator relies on the force that exists between the poles of magnets. Reluctance torque is a consequence of the

principle of conservation of energy, and arises due to the fact that the most stable configuration of a magnetic system is that of minimum reluctance. Magnetostriction causes generation of very high forces in ferromagnetic materials when subjected to applied magnetic fields. We show that a vibrating beam can be stabilized if the magnetizing current in the magnetic actuator is varied proportional to the rate of change of beam bending moment or the beam tip angular velocity. These magnetic actuators can be implemented using ferromagnetic or ferroplastic materials, and can be applied over the entire spatial domain of the elastic structure, thus emulating a distributed control actuator. Since the energy density of a magnetic device is much higher compared to piezo-electric or thermal actuators, it is expected that the magnetic actuator will be more effective in controlling the structural vibrations.

II. MAGNETIC ACTUATORS

A magnetomechanical transducer or actuator is a device that links a magnetic system and a mechanical system. The coupling between the two systems is through the magnetic field which acts as the energy storage device. A change in the stored energy leads to a energy conversion process to convert the magnetic energy to the mechanical energy, or vice-versa. There are several electromagnetic phenomena [22,23] that govern this energy conversion process among which the following are most important, and are commonly utilized in practical devices:

1. A mechanical force is exerted on a current carrying conductor in a magnetic field. Likewise, mechanical forces exist between two current carrying conductors because of their own magnetic fields.

2. A mechanical force is exerted on a movable ferromagnetic material tending to align it along the magnetic flux lines, or to reduce the reluctance of the flux path.

3. Most ferromagnetic materials show a small deformation in the presence of a magnetic field. This phenomenon is known as *magnetostriction*. Although the deformation is very small, the corresponding mechanical force may be very large.

All the above energy conversion processes are reversible in the sense that applications of mechanical forces or body deformations produce changes in the magnetic energy. In this research, we intend to utilize the magnetic-to-mechanical energy conversion processes for production of forces for stabilization of structural vibrations of elastic systems. In what follows, we present the fundamentals of three magnetic actuators which have very good potential of practical implementation for stabilization of large flexible space structures.

2.1 MAGNETIC BODY-FORCE ACTUATOR

The basic idea of this device is the magnetic body-force or stress acting between the magnetic poles. Consider the attraction of north and south poles of two magnets. The total force on one pole face is given by the integration of magnetic stress as

$$F_m = \int_A \frac{B_n^2}{2\mu_0}\, dA \tag{1}$$

where B_n is the normal component of the field density to the surface, and μ_0 is the permeability of the air gap. Consider a magnetic system consisting of two ferromagnetic elements separated by a distance, and with I_1 and I_2 as the magnetizing currents as shown in Fig. 1.

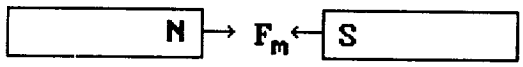


Fig. 1 Magnetic Body-Force

Then it can be shown that the resultant magnetic body-force is given by

$$F_m = k_1 I_1 I_2. \tag{2}$$

We may assume that one of the ferromagnetic materials is replaced by a permanent magnet, or an electro magnet with a constant exciting current. Then the force resulting from this magnetic system is of the form

$$F_m = k_2 I \tag{3}$$

where k_2 is suitable constant, and I is the magnetzing current. This analysis shows that this simple configuration of magnetic materials may be used for production of a force, and that this force could be made proportional to a control current. For the sake of simplicity, we assume that the magnetic force is distributed all over the spatial domain. In fact, for a single layer of ferromagnetic segments, this force may appear as a train of step functions. By using several layers of segments, one can obtain an average force that is distributed all over the spatial domain.

Now consider a flexible beam with a layer of ferromagentic segments rigidly attached to the upper surface of the beam, and another layer on the lower surface as shown in the Fig. 2.

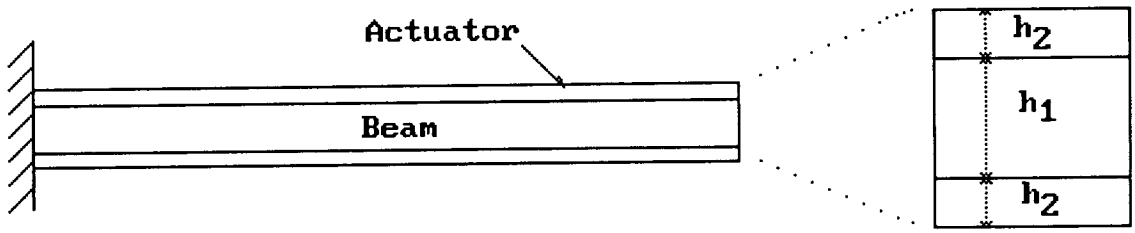


Fig. 2 Flexible Beam with Body-Force Actuator

We assume that the same magnetizing current is used for both the upper and the lower layers, and that the corresponding forces are same in magnitude but opposite in direction. This results in a bending moment given by

$$\begin{aligned} T(x,t) &= F_m(x,t)(h_1 + h_2) \\ &= c\, I(x,t) \end{aligned} \tag{4}$$

where c is a constant depending of the beam geometry and the properties of the magnetic material.

The dynamics of the transverse vibrations of a beam in the presence of this additional bending moment is given by

$$\rho \frac{\partial^2 y}{\partial t^2} + \frac{\partial^2}{\partial x^2}\left(Y \frac{\partial^2 y}{\partial x^2}\right) - \frac{\partial^2}{\partial x^2}\Big(c\, I(x,t)\Big) = 0, \quad x \in (0,L), \quad t \geq 0 \tag{5}$$

with the boundary conditions

$$\begin{aligned} y(0,t) &= 0 & Y\frac{\partial^2 y}{\partial x^2}(L,t) &= c\, I(L,t) \\ \frac{\partial y}{\partial x}(0,t) &= 0 & \frac{\partial}{\partial x}\left(Y\frac{\partial^2 y}{\partial x^2}\right)(L,t) &= c\,\frac{\partial I}{\partial x}(L,t) \end{aligned} \tag{6}$$

where y is the transverse deflection, Y is the flexural rigidity, and ρ is the mass density (per unit length) of the composite beam.

STABILIZATION

The beam dynamics described above contains a controllable parameter $I(x,t)$ which may be appropriately regulated in order to achieve a stabilizing action. For this purpose we follow the Lyapunov type analysis. Consider the total energy of beam vibrations given by

$$V(t) = \frac{1}{2}\int_0^L \left\{ \rho \left| \frac{\partial y}{\partial t} \right|^2 + Y \left| \frac{\partial^2 y}{\partial x^2} \right|^2 \right\} dx. \tag{7}$$

Then using the dynamics (5) along with the boundary conditions, we obtain

$$\frac{dV}{dt} = \int_0^L c\, I(x,t)\, \frac{\partial^3 y}{\partial x^2\, \partial t}\, dx. \tag{8}$$

This clearly shows that for asymptotic decay of vibration energy the magnetizing control current may be chosen as

$$I(x,t) = -k\, \frac{\partial^3 y}{\partial x^2\, \partial t}(x,t) \tag{9}$$

where k is a suitable gain, in other words, the control current should be proportional to the rate of change of bending moment of the beam.

It is interesting to note that this ferromagnetic actuator essentially introduces in the system a type of damping commonly known as "*structural damping*" in the literature. Indeed, substituting the equation (9) into the dynamics (5), the beam equation can be rewritten as

$$\rho \frac{\partial^2 y}{\partial t^2} + \frac{\partial^2}{\partial x^2}\left(Y \frac{\partial^2 y}{\partial x^2} \right) + k \frac{\partial^5 y}{\partial x^4\, \partial t} = 0, \tag{10}$$

in which the last term represents the structural damping. Note that the damping parameter k is very small for naturally occurring structural damping of elastic materials. In this case the control current can be suitably regulated so as to obtain the desired damping.

In case the feedback current is assumed to be uniform all over the length of the beam, equation (8) reduces to

$$\frac{dV}{dt} = c\, I(t)\, \frac{\partial^2 y}{\partial x\, \partial t}. \tag{11}$$

Hence considering a feedback current proportional to the tip angular velocity of the beam, i.e.,

$$I(t) = -k \frac{\partial^2 y}{\partial x\, \partial t}(L,t) \tag{12}$$

we obtain asymptotic stability of the system.

The control laws discussed above require regulation of the control current proportional to the angular velocity of the tip of the beam, or the rate of change of the (distributed) bending moment. For practical applications it may be relatively easier to measure the tip angular velocity only. Proportional variations of the control current can be done by suitable electronic circuits. One can also consider on-off or deadzone type of controls derived [9] from (8).

2.2 RELUCTANCE ACTUATOR

A property of a conservative system is that its energy is a function of only its state, and given sufficient time, the system always attains its rest state at which the energy is minimum. Consider a magnetic circuit containing a movable member. The energy stored in the magnetic field is minimum when the movable member attains a position for which the magnetic reluctance is minimum. Any perturbation of this position would imply a higher energy state of the system, and hence would lead to the production of a restoring force or torque that will realign the movable member to the minimum reluctance position. This is the fundamental principle of the Reluctance Actuator.

Consider a magnetic circuit consisting of two ferromagnetic segments as shown before; but in this case we assume that these segments can undergo an angular displacement relative to each other.

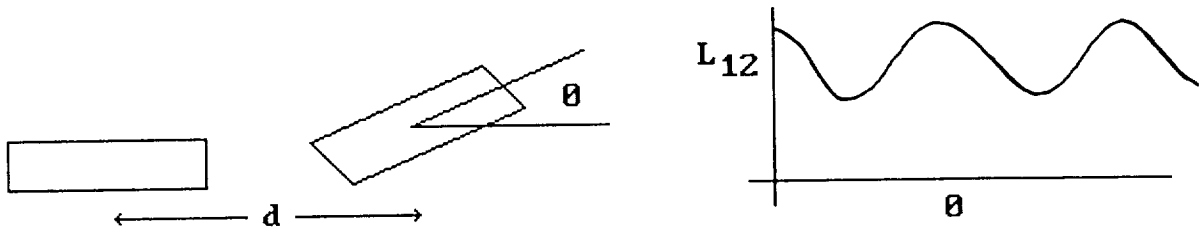


Fig. 3 Variation of Mutual Inductance with Angular Position

The magnetic potential energy stored in the air gap depends on the mutual inductance and the magnetizing currents, and is given by

$$W_m = L_{12}\, I_1\, I_2. \tag{13}$$

The mutual inductance L_{12} varies with the angular orientation of the two segments relative to each other. It is clear from Fig. 3 that when θ is 0° or 360°, reluctance is minimum so that inductance is at the maximum value. Similarly, when θ is 180°, reluctance is maximum with the correspondingly small inductance. Hence the mutual inductance can be expressed as

$$L_{12} = L_0 + L_\sigma \cos\theta. \tag{14}$$

Any rotation of the movable member would tend to increase the air gap, and hence would increase the reluctance, or decrease the inductance. Then according to the principle of conservation of energy, a restoring torque is produced that would realign the movable member with the stationary member. This restoring torque is given by

$$\begin{aligned} T &= \frac{\partial W\,(I_1, I_2, \theta)}{\partial \theta} \\ &= -L_\sigma\, I_1\, I_2 \sin\theta. \end{aligned} \tag{15}$$

Clearly, the torque reduces to zero when there is no angular deflection, i.e., when the two segments are aligned. In what follows, we show that this torque can be utilized to stabilize a vibrating beam.

Consider a cantilever beam with a string of ferromagnetic segments interlaced by air gaps as shown in the Fig. 2. Consider two typical segments located at the axial distances $x - \frac{d}{2}$ and $x + \frac{d}{2}$ respectively, where d is the distance between the two segments. The angular orientation of these segments on a perturbed beam will be given by $\frac{\partial y}{\partial x}(x - \frac{d}{2})$ and $\frac{\partial y}{\partial x}(x + \frac{d}{2})$ respectively. Hence the relative angle between the two segments is

$$\begin{aligned} \theta(x,t) &= \frac{\partial y}{\partial x}(x + \frac{d}{2}) - \frac{\partial y}{\partial x}(x - \frac{d}{2}) \\ &\simeq d\,\frac{\partial^2 y}{\partial x^2}(x,t) \end{aligned} \tag{16}$$

Using this equation in (15) and assuming small angle perturbations of the beam, the restoring torque becomes

$$T(x,t) = -L_\sigma I_1 I_2 \sin\left(d\frac{\partial^2 y}{\partial x^2}\right)$$
$$\simeq -d L_\sigma I_1 I_2 \frac{\partial^2 y}{\partial x^2}(x,t) \tag{17}$$

For simplicity we assume that the magnetizing currents are equal, and $I_1 = I_2 = I$. Then the dynamics of transverse vibration of the beam is given by

$$\rho\frac{\partial^2 y}{\partial t^2} + Y\frac{\partial^4 y}{\partial x^4} + kI^2\frac{\partial^4 y}{\partial x^4} = 0 \tag{18}$$

with appropriate boundary conditions. This shows that the reluctance torque essentially increases the flexural rigidity of the elastic material, and this stiffening action is independent of the direction of the magnetizing current. Thus reluctance torque can be used to introduce artificial flexural rigidity in elastic members. Alternatively, feedback control schemes can be designed to stabilize the system. Indeed, after some analysis using the energy function (7), it can be shown that a feedback current proportional to the rate of change of bending moment of the elastic member,i.e.,

$$I_1 I_2(t) = -q\frac{d}{dt}\left\|\frac{\partial^2 y}{\partial x^2}\right\|^2_{L_2} \tag{19}$$

can be used to stabilize the system. Here q is the gain of the controller.

2.3 MAGNETOSTRICTION

Magnetostriction is the elastic deformation of a magnetic material due to the change in the magnetic field. If a ferromagnetic bar such as nickel, cobalt, is subjected to an applied magnetic field, it shrinks in length. If the bar is restrained from contracting, a mechanical force is developed and mechanical energy can be extracted. For some magnetic materials the action is to elongate rather than contract while in some others first to elongate and then contract. The change in length is usually very small and of the order of 0.01%, but the resulting force may be very large of the order of 200 N/cm^2 or 300 psi. It is important to note that the stress due to magnetostriction is independent of the direction of the applied magnetic field. As such the mechanical force obtainable from a magnetostrictive device will be bounded between zero and some upper limit depending on the strength of the applied field.

III. CONCLUSIONS

We consider stabilization of flexible structures using three types of magnetomotive forces: a) magnetic body-force, b) reluctance torque, and c) magnetostriction. We prove stabilization of the system using the first two types of forces. This requires feedback of rate of bending moment of the structure in the form of a magnetizing current. It is important to note that magnetic body-force and the reluctance torque are complementary and occur simultaneously, in other words, the same hardware will produce two types of stabilizing action in the vibrating system. Although magnetostriction produces a mechanical force that can be extracted, at this time it is not clear whether this can be utilized to produce any stabilizing action.

IV. BIBLIOGRAPHY

1. Bodley C.S. and Park A.C., "The Influence of Structural Flexibility on the Dynamic Response in Spinning Spacecraft", *AIAA Paper 72-348*, San Antonio, Texas, 1972.

2. Gale A.H. and Likins P.W., "Influence of Flexible Appendages on Dual-Spin Spacecraft Dynamics and Control", *Journal of Spacecraft and Rockets*, Vol. 7, pp. 1049-1056, 1970.

3. Grote P.B., McMunn J.C., and Gluck R., "Equations of Motion of Flexible Spacecraft", *Journal of Spacecraft and Rockets*, Vol. 8, pp. 561- 567, 1971.

4. Hookes W.W. and Margelics G., "The Dynamical Attitude Equations for an n-body Satellite", *Journal of Astronautical Sciences*, Vol. 12, no. 4, 1965.

5. Sesak J., Likins P. and Corodeth T., "Flexible Spacecraft Control by Modal Error Sensitivity", *Journal of Astronautical Sciences*, Vol. 27, pp. 131-156, 1979.

6. Breakwell, J.A. "Optimal Feedback Slewing of Flexible Spacecraft", *Journal of Guidance and Control*, Vol. 4, pp. 472-479, 1981.

7. Hughes P. and Skelton R., "Controllability for Flexible Spacecraft", *Journal of Guidance and Control*, Vol. 3, pp. 452-459, 1980.

8. Biswas, S.K. and Ahmed, N.U., "Stabilization of a Class of Hybrid Systems Arising in Flexible Spacecraft", *Journal of Optimization Theory and Applications*, Vol. 50, pp. 83-108, July, 1986.

9. Biswas S.K. *Modeling and Stabilization of Flexible Spacecraft*, Ph.D. dissertation, Department of Electrical Engineering, University of Ottawa, 1985.

10. Araya, R., "A Controllability-Stabilizability Result for the SCOLE", *Proceedings of the NASA SCOLE Workshop*, Langley Research Center, Hampton, Virginia, December, 1984.

11. Balakrishnan, A.V., "A Mathematical Formulation of the SCOLE Control Problem", *NASA Report*, NASA CR-17258, Langley Research Center, Hampton, Virginia, 1985.

12. Taylor, L. and Balakrishnan, A.V., "A Mathematical Problem and a Spacecraft Control Laboratory Experiment (SCOLE) used to Evaluate Control Laws for Flexible Spacecraft --- NASA/IEEE Design Challenge", *Proc. SCOLE Workshop*, NASA Langley Research Center, Hampton, 1984.

13. Meirovitch, L., "Stability of a Spinning Body Containing Elastic Parts via Lyapunov's Direct Method", *AIAA Journal*, Vol. 8, pp. 1193-1200, 1970.

14. Meirovitch, L., "A Method of Lyapunov Stability Analysis of Force-Free Dynamical Systems", *AIAA Journal*, Vol. 9, pp. 1695-1700, 1971.

15. Balas M.J., "Trends in Large Space Structure Control Theory: Fondest Hopes and Wildest Dreams", *IEEE Trans. on Automatic Control*, Vol. AC-27, pp. 522-535, 1982.

16. Biswas S.K., "Optimal Regulation of Flexible Structures Governed by Hybrid Dynamics", *Proceedings of the 27th IEEE Conference on Decision and Control*, Austin, Dec 6-9, 1988, pp. 1613-1618.

17. Ashley H., "On Passive Damping Mechanism in Large Space Structures", *Journal of Spacecraft and Rockets*, Vol. 21, pp. 448-455, 1984.

18. Haftka R.T. and Adelman H.M., "An Analytical Investigation of Shape Control of Large Space Structures by Applied Temperatures", *AIAA Journal*, Vol. 23, pp. 450-457, 1985.

19. Edberg D.L., "Applying Thermal Gradients to Control Vibrations", JPL report No. NPO-17067/6575, 1989.

20. Bailey Thomas and Hubbard James E., "Distributed Piezoelectric-Polymer Active Vibration Control of a Cantilever Beam", *Journal of Guidance*, Vol. 8, pp. 605-611, 1985.

21. Burke S. and Hubbard J.E., "Active Vibration Control of a Simply Supported Beam using a Spatially Distributed Actuator", *IEEE Control Systems Magazine*, Vol. 7, pp. 25-30, 1987.

22. Nasar S.A., *Electromagnetic Energy Conversion Devices and Systems*, Prentice Hall, Englewood Cliffs, 1970. McGraw Hill, 1974.

23. Moon F. *Magnetosolid Mechanics*, John Wiley, New York, 1984.

Optimal Control of Systems with Capacity-related noises
Abstract

Mifang Ruan* and Ajit K. Choudhury+

In the ordinary theory of optimal control (LQR and Kalman Filter), the variances of the actuators and the sensors are assumed to be known(not related to the capacities of the devices). This assumption is not true in practice. Generally , a device with greater capacity to exert actuating forces and a sensor capable of sensing greater sensing range will generate noise of greater power spectral density.

When the ordinary theory of optimal control is used to estimate the errrors of the outputs in such cases it will lead to faulty results, because the capacities of such devices are unknown before the system is designed. The performance of the system designed by the ordinary theory will not be optimal as the variances of the sensors and the actuators are neither known nor constant. The interaction between the control system and structure could be serious because the ordinary method will lead to greater feedback (Kalman gain) matrices.

The main purpose of this paper is to dvelop methods which can optimize the performance of systems when noises of the actuators and the sensors are related to their capacities. These methods will result in smaller feedback (Kalman gain) matrix. The smaller matrices will reduce the interaction between the control system and system structure and, thereby, reducing the requirements on the structures and consequently making the structure more flexible.

INTRODUCTION

In the optimal control of stochastic systems, we ordinarily assume that noises of the actuators and the sensors are not related to the capacity of the actutors and sensors[1,2,3] This assumption is not true in practice.Generally, the variances of actuators and the sensors, especially the actuators , are related to the capacities of the devices. Obviously, a fuel jet capable of generating a force of 100 lbs will have greater noises than the one capable of generating a force of 1 lb. It will be realistic and practical to assume that the noise variance of the actuators and the sensors be linear function of the variance of the controlling forces and the output of the sensors i.e., the observations. Under this assumption when a device is required to have greater capacity it will also introduce greater noise. The ordinary method of optimal control problems have at least three defects.

(a) It is hard to specify correctly the noise power spectral densities of the actuator and the sensor because the capacities of these devices are unknown

* Graduate Student, Dept. of Mechanical Engr., Howard Univ., Washington, D.C. 20059,+ Assoc. Professor, Dept. of Electrical Engineering, Howard Univ.

before the system is designed

(b) The resultant feedback and Kalman gain matrices may not be optimal when the noises in these devices are not related to their capacities. Therefore, the performance of the control system may not be as good as it would be otherwise.

(c) When the noises are assumed to be not related to the capacities of the devices, the resultant feedback and Kalman gain matrices will be large, making the interaction between control system and the structure unsafe[4].

Because of these defects, it is hard for us to estimate the errors of the outputs. The errors of the outputs will be large, and the interaction between the control system and system structures will produce large errors.

In some control systems, such as communications satellite and on-orbit telescope, the precision of the control system is critical, and in the future missions their structures could be very flexible. The variances of these devices will be assumed to be linear functions of their capacities.

In this paper, we will develop methods which will optimize the performance of systems when noises of the actuators and the sensors are related to their capacities. The feedback (and Kalman gain) matrices are found by this methods will be automatically smaller than those found by ordinary methods. Therefore, the interaction between the control system and the strurcture will be reduced and thereby, permitting more flexible structures.

II PROBLEM STATEMENT

Let us first consider the optimal control of a first order system

$$\dot{x} = a x + u + w \tag{1.a}$$

$$u = -f x \tag{1.b}$$

$$J = E \left\{ x^2 + r u^2 \right\} \tag{1.c}$$

This is a steady-state optimal control problem with exact observation. f is the feedback coefficient to be determined. a and r are given parameters. E is the mean operator. w is a zero mean white Gaussian noise. Unlike the ordinary control problem, we assume that the variance of the noise w, can be described by

$$W = W_0 + \alpha \sigma_u^2$$

$$= W_0 + \alpha f^2 \sigma_x^2$$

where W_0 , and α are non-negative constants. σ_u^2 can be considered as the nominal variance of the input, a good measure of the capacity of the actuator.From (1a) and (1b) , we have

$$\dot{x} = (a - f)x + w \tag{3}$$

According to stochastic control theory, the variance of x, denoted by P can be determined by

$$2(a-f)P + W = 0 \tag{4}$$

Since $\sigma_x^2 = P$, eq.(4) reduces to

$$2 (a - f) + W_0 + \alpha f^2 P = 0 \tag{5}$$

$$\text{or} \quad (2a - 2f + \alpha f^2) P + W_0 = 0 \tag{6}$$

Since P must be greater than or equal to zero, the following condition must hold

$$2a - 2f + \alpha f^2 < 0 \tag{7}$$

The cost functional can be written as

$$J = P + r f^2 P \tag{8}$$

Using eq.(6), we have

$$J = (1 + r f^2) \frac{W_0}{(2 f - 2 a - \alpha f^2)} \tag{9}$$

The inequality above indicate the stable region for the problem.

The stable region for ordinary problem is defined by

$$a - f < 0 \qquad (10)$$

The stable regions describe by equations (7) and (10) are plotted in Fig. 1. Obviously, the stable region of the present problem is only a subset of the region of the ordinary problem. The stable region becomes smaller when α becomes greater and this region is not directly related to the constant term. For certain values of $a > 0$ and α, it is possible that there is no f which lie in the stable region, i.e., such a system can't be stabilized.

The optimal feedback control can be found by differentiating equation (9) with respect to f and equating the derivative of J with respect to f to zero. The derivative of J with respect to f after simplification can be written in the form

$$\frac{dJ}{df} = \frac{2W_0}{(2f - 2a - \alpha f^2)^2} \left(rf^2 - 2ra f + \alpha f - 1 \right)$$

Equating the derivative of J to zero and solving the quadratic equation in f and neglecting the extraneous solution, we obtain the optimal feedback control as follows.

$$f = \frac{-(\alpha - 2ra) + \sqrt{(\alpha - 2ra)^2 + 4r}}{2r} \qquad (11a)$$

Figs 2,4 and 6 show the ratio of optimal feedback (the value of f given by equation (11a)) to the feedback found by ordinary method vs. r for various values of a and α. We can see that the value of f/fo is less than 1, i.e., when the noise of the actuator is capacity related, the optimal feedback tends to decrease.The reason for this is that a greater feedback corresponds to a greater actuator signal, and increased capacity of the device and increased noise power spectral density of the noise.Therefore, a smaller feedback matrix will be preferred. When r becomes smaller, α becomes bigger and a becomes greater. The difference between f and f_0 will becomes greater. The reason is not hard to imagine. When r becomes smaller, the feedback by the ordinary method becomes greater even it is out of the stable region, while the feedback by the present method although becomes greater but the increment will not be significant because it has not taken the increase in noise power into consideration, and the feedback will never be out of the stable region. When α (alpha) becomes greater, noise is more related to the capacity and the system will more seriously depend on the feedback. Figs 3,5 and 6 show the ratios of optimal cost found by the propsed method to the case when f is found by the ordinary method. Fig.5 does not have a plot for $\alpha = 1$, because the feedback found by the ordinary method is out of stable region, and the ratios, Jo/J is infinite.

III GENERRAL CASE

In the above section, we have solved a simple problem by theoretical approach. In general system contains multiple states with multiple inputs and multiple outputs, and the meassurements are corrupted by noises. Then , the problem can be stated as follows.

$$\min_{F,K} \quad J = E \{ y^T Q y + u^T R u \} \tag{11b}$$

with constraints

$$y = H x$$

$$\dot{x} = A x + B u + G w \tag{12.a}$$

$$u = - F \hat{x} \tag{12.b}$$

$$\dot{\hat{x}} = A \hat{x} + B u + K (z - M \hat{x}) \tag{12.c}$$

where

y= Output vector
x= State vector
u= Control Vector
z= Measurement Vector
x= Estimated State Vector
w= input noise vector (zero–mean white Gaussian noise)
v= Measurement noise vector(zero mean white Gaussian noise)

F = Feedback matrix
K = Kalman Gain Matrix
and the matrices are of appropriate dimension

The noise covariance matrices are given by

$$E \{ w \quad w^T \} = W$$

$$E \{ v \quad v^T \} = V$$

The only difference between the ordinary problem and proposed problem is that the model of the covariance matrices w and v. Ordinarily W and V are assumed to be constant matrices which are not related to the capacities of the

actuators and sensors. In this paper W and V are assumed to be matrices whose covariance matrices are functions of the capacities of actuators and sensors.

The capacity of an actuator can be reasonably be represented by the nominal varianace of the actuator signal,

$$\sigma_{u_i}^2 = E_{\infty} \{u_i^2\} = f_i E_{\infty} \{ \hat{x} \hat{x}^T \} f_i^T \triangleq f_i P_{\hat{x}} f_i^T$$

where f_i is the ith row of F.

And , we will assume the variance of an actuator to be a linear function of its capacities , i.e.,

$$W = \text{diag} \{ w_1 , w_2 , \cdots \cdots w_m \}$$

$$w_i = w_{i_0} + \alpha_i \sigma_{u_i}^2$$

where w_{i_0} and α_i are non–negative constants

Similarly,

$$V = \text{diag} \{ V_1 , V_2 , \cdots \cdots V_l \}$$

$$V_i = V_{i_0} + \beta_i \sigma_{z_i}^2$$

where V_{i_0} and β_i are non – negative constants

$$\sigma_{z_i}^2 = E \{ z_i^2 \} = m_i E \{ x x^T \} m_i \triangleq m_i P_x m_i^T$$

where m_i is the ith row of M.

Clearly the covarince matrices become functions of the feedback and Kalman gain matrices.

Equation (12.a) ,(12.b) and (12.c) can be written as

$$\begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \end{bmatrix} = \begin{bmatrix} A & -BF \\ KM & A - BF - KM \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix} + \begin{bmatrix} G & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix}$$

According to the Stochastic Control theory the covariance matrices satisfy the following equations

$$\begin{bmatrix} P_x & P_{x\hat{x}} \\ P_{x\hat{x}} & P_{\hat{x}} \end{bmatrix} \begin{bmatrix} A & -BF \\ KM & A - BF - KM \end{bmatrix}^T + \begin{bmatrix} A & -BF \\ KM & A - BF - KM \end{bmatrix} \begin{bmatrix} P_x & P_{x\hat{x}} \\ P_{x\hat{x}} & P_{\hat{x}} \end{bmatrix} + \begin{bmatrix} G & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} W & 0 \\ 0 & V \end{bmatrix} \begin{bmatrix} G & 0 \\ 0 & K \end{bmatrix}^T = 0$$

where W and V are functions of P_x , $P_{\hat{x}}$, F and K .To solve the above optimization problem, we probably have to use numerical approach

IV . DIRECT APPROACH

The simplest way to solve the problem is to use direct approach. In the direct approach, we assume that all the elements of F and K are parameters. The cost J can be found by solving equation (11b) iteratively when F and K are given. Various techniques of optimization theory can be used to find the optimum value of F and K.

However, this method can solve only problems of smaller dimension.

For relatively large problems, the number of parameters will be large and the computational efforts to find the cost for given F and K will also be large;therefore, the total computational load will be large

It seeems that the challenging problem here is development of computationally efficient fast algorithm to sove the feedback gain and the Kalman gain

REFERENCES

1. Skelton, R.E., and Delorenzo, M , "Space Structure Control Design by Variance Assignment, " J. Guidance . vol.8, no.4,1985, pp. 454-62

2. Kwakernack, H., and R. Sivan, Linear Optimal Control Systems, Wiley-Interscience, 1972, New York.

3. Mayheck, P.S., Stochastic Models, Estimation and Control,Academic Press, 1979,New York.

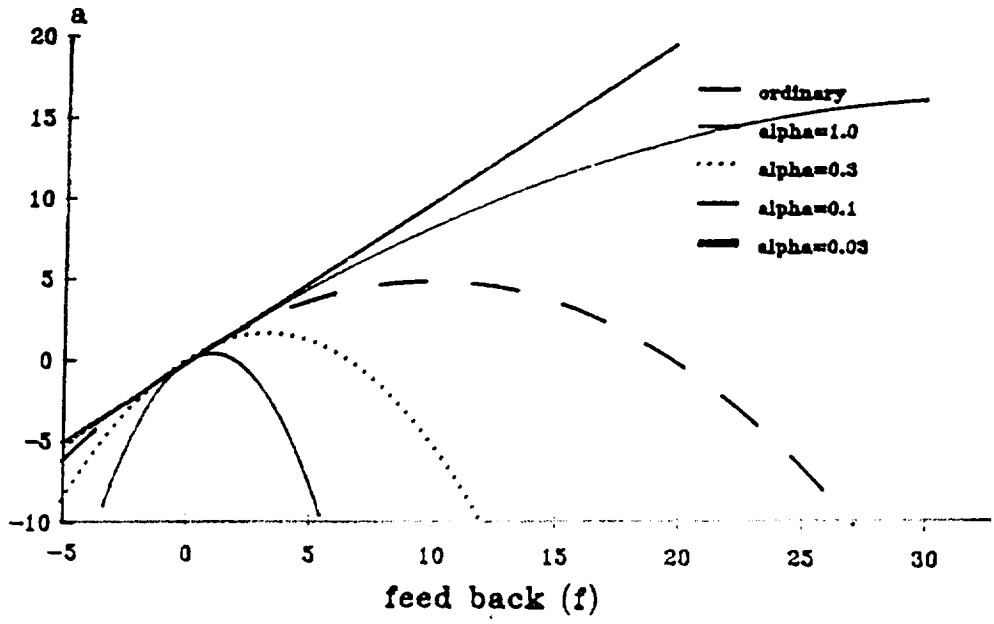

Optimal Contr. with Cap-Related Noise
stable regions
a
20
15
10
5
0
-5
-10
-5
0
5
10
15
20
25
30
feed back (f)
ordinary
alpha=1.0
alpha=0.3
alpha=0.1
alpha=0.03

Fig.1

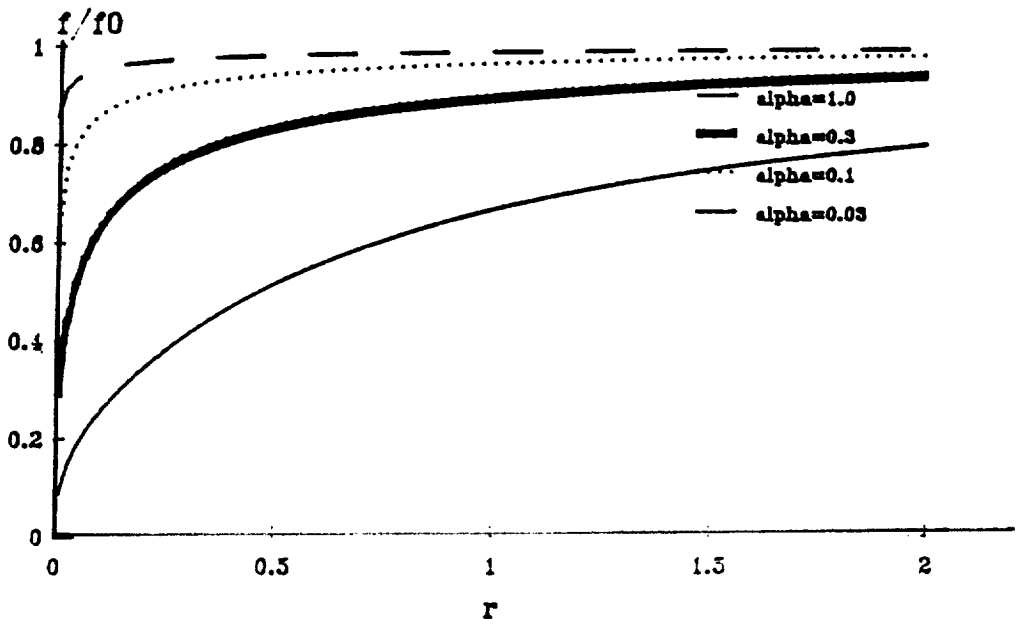

Comparison of the Feedback
a=1.
f/f0
1
0.8
0.6
0.4
0.2
0
0
0.5
1
1.5
2
r
alpha=1.0
alpha=0.3
alpha=0.1
alpha=0.03

Fig.4

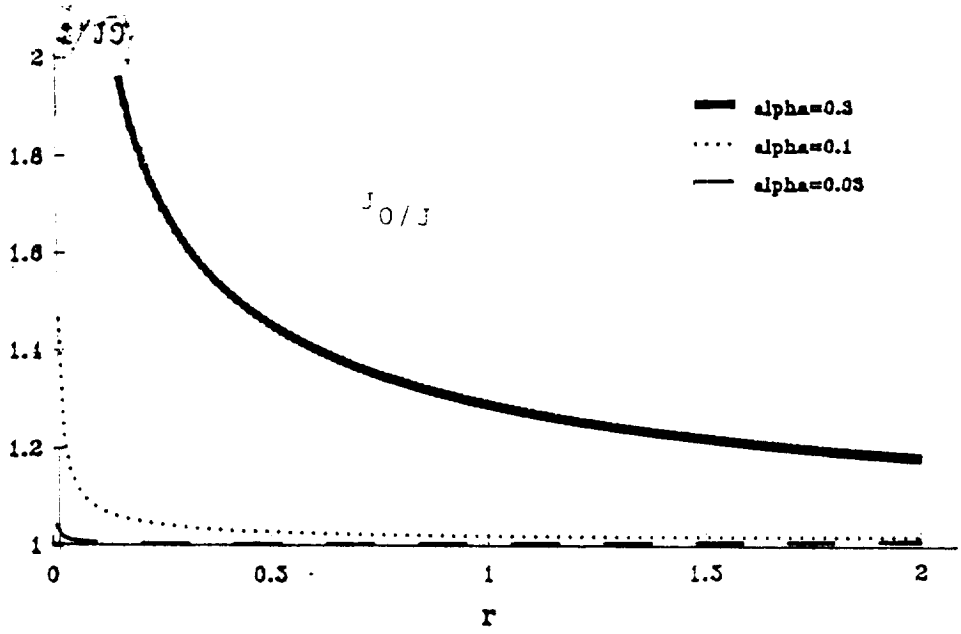
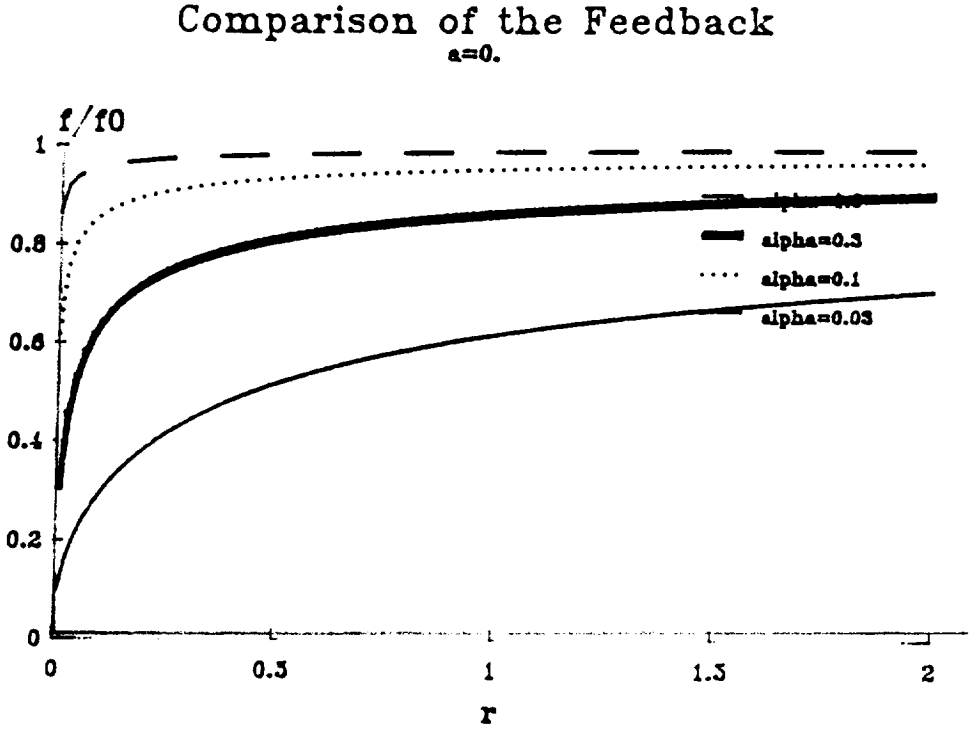

Comparison of the Total Cost
a=1.
J/J0
2
1.8
1.6
1.4
1.2
1
0
0.5
1
1.5
2
r
J0/J
alpha=0.3
alpha=0.1
alpha=0.03

Fig.5


Comparison of the Feedback
a=0.
f/f0
1
0.8
0.6
0.4
0.2
0
0
0.5
1
1.5
2
r
alpha=0.3
alpha=0.1
alpha=0.03

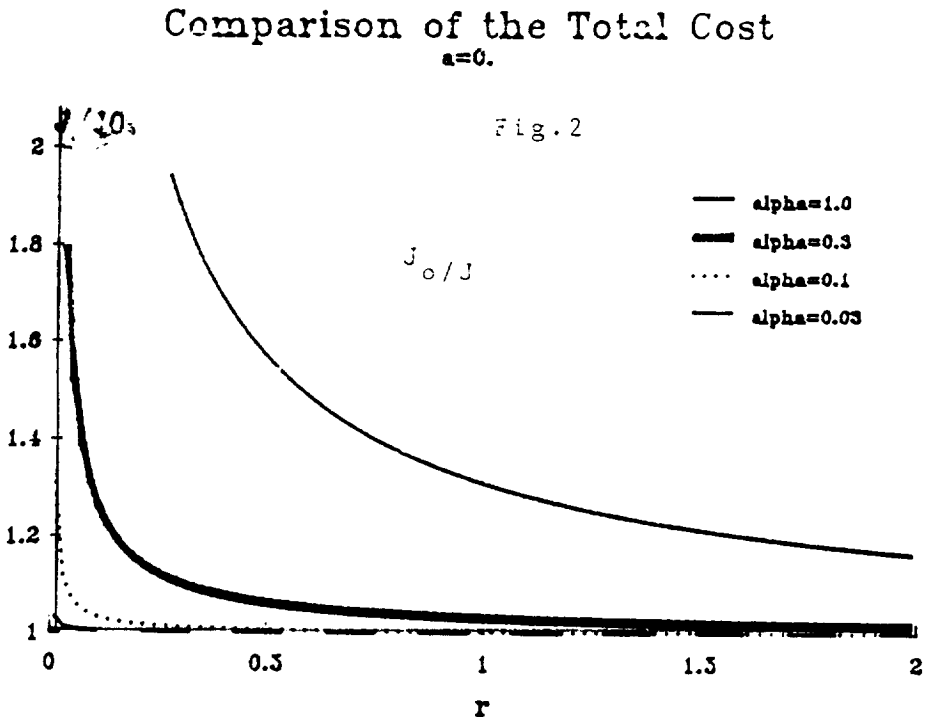

Comparison of the Total Cost
a=0.
2
1.8
1.6
1.4
1.2
1
0
0.5
1
1.5
2
r
alpha=1.0
alpha=0.3
alpha=0.1
alpha=0.03

Fig.2

Fig.3

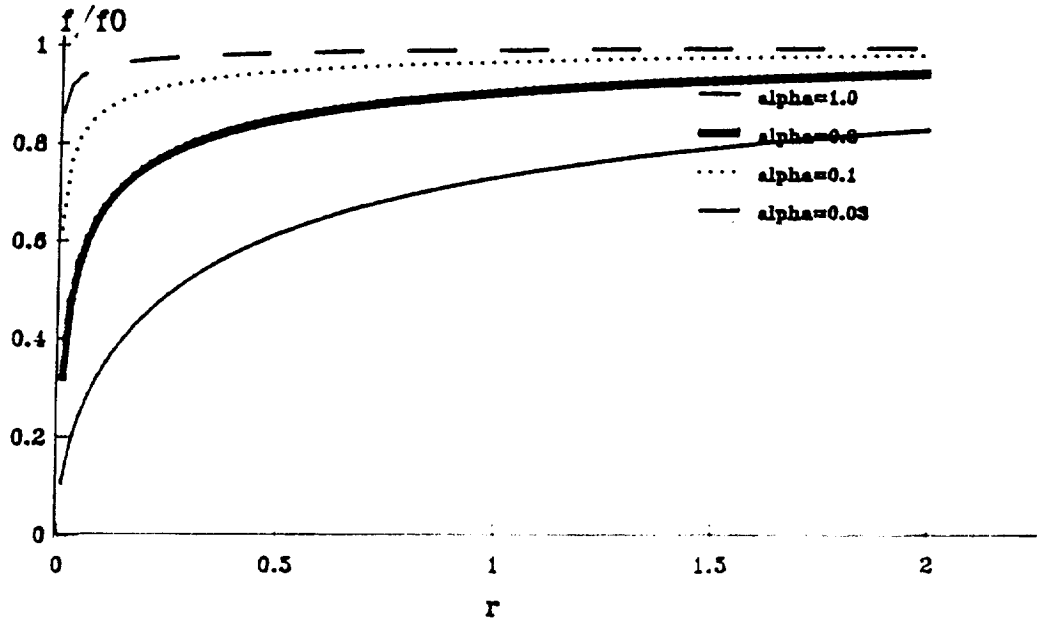

Comparison of the Feedback
a=-1.
f/f0
1
0.8
0.6
0.4
0.2
0
0
0.5
1
1.5
2
r
alpha=1.0
alpha=0.3
alpha=0.1
alpha=0.03

Fig.6

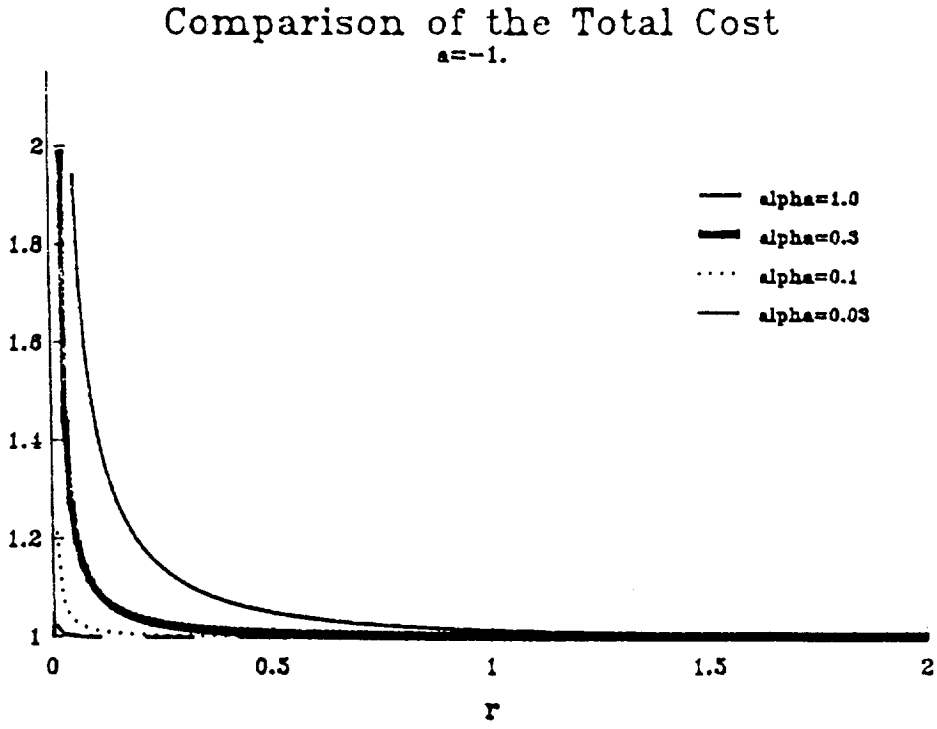

Comparison of the Total Cost
a=-1.
2
1.8
1.6
1.4
1.2
1
0
0.5
1
1.5
2
r
alpha=1.0
alpha=0.3
alpha=0.1
alpha=0.03

Fig.7

Querying Databases of Trajectories of Differential Equations II: Index Functions

Robert Grossman*
University of Illinois at Chicago

June, 1990

Abstract Suppose that a large number of parameterized trajectories γ of a dynamical system evolving in $\mathbf{R}^N$ are stored in a database. Let $\eta \subset \mathbf{R}^N$ denote a parameterized path in Euclidean space, and let $||\cdot||$ denote a norm on the space of paths. In this paper, we define data structures and indices for trajectories and give algorithms to answer queries of the following forms:
Query 1. Given a path η, determine whether η occurs as a subtrajectory of any trajectory γ from the database. If so, return the trajectory; otherwise, return null.
Query 2. Given a path η, return the trajectory γ from the database which minimizes the norm

$$||\eta - \gamma||.$$

1 Queries about trajectories

Suppose that a large number of parameterized trajectories γ of a dynamical system evolving in $\mathbf{R}^N$ are stored in a database. Let $\eta \subset \mathbf{R}^N$ denote a parameterized path in Euclidean space, and let $||\cdot||$ denote a norm on the space of paths to be specified later. In this paper, we define a data structure and indices to represent trajectories of dynamical systems and sketch algorithms to answer queries of the following forms:
Query 1. Given a path η, determine whether η occurs as a subtrajectory of any trajectory γ from the database. If so, return the trajectory; otherwise, return null.
Query 2. Given a path η, return the trajectory γ from the database which minimizes the norm

$$||\eta - \gamma||.$$

The paper is a successor to [2], which describes the data structure to store trajectories which is used here.

Efficient algorithms to answer these type of queries should prove useful for a number of applications. As an example, consider the path-planning problem for a robotic arm. Suppose that a large number of feasible trajectories of the robotic arm have been stored in a database. Let η be the desired path of the arm. It is not necessary that η itself be a feasible trajectory. Query 2 would return the feasible trajectory γ of the arm which is closest to the desired path η.

As another example, consider a database containing control trajectories for an aircraft. Assume that those trajectories which enter into an unstable control regime somewhere along their flight path are tagged.

*This research is supported in part by grant NAG2-513 from NASA and by grant DMS-8904740 from the National Science Foundation and by the Laboratory for Advanced Computing. Address: Department of Mathematics, Statistics, and Computer Science, Mail Code 249, University of Illinois at Chicago, Box 4348, Chicago, IL 60680, (312) 413-2164, grossman@uicbert.eecs.uic.edu.

Let η denote a measured portion of the flight path. Then Query 1 would return the nearest full control trajectory in the database, which includes information about the stability of the trajectory. More generally, one could imagine retrieving from the database those stable trajectories which avoid a given obstacle, such as a turbulent region of space. In other words, the query could be used as part of a supervisory control system and be viewed as a means of extracting qualitative or summary information about the control system.

The data structure we use to represent trajectories is closely related to hashing methods for curves that have been used in computer vision; see [7] and [8]. A related means of extracting qualitative information from dynamical systems is described in [1]. We are concerned in this paper with data structures and indexing for object oriented databases consisting of trajectories. For general methods of indexing in object oriented databases see [3], [4], [5], and [6].

In Section 2 we review the relevant facts about trajectories of differential equations and define different data structures to store trajectories. In Section 3, we show how these data structures can be used to answer the queries above. Section 4 contains some concluding remarks.

2 Paths, trees and vector fields

In this section, we describe a data structure for paths following [2]. The point of view is to assume that the path arises from a trajectory of a differential equation and to base the data structure upon the initial value problem for the differential equation.

We begin by recalling some basic facts and definitions about trajectories of differential equations. Let $D_\mu = \partial/\partial x_\mu$. A *vector field*

$$E = \sum_{\mu=1}^{N} a^\mu D_\mu$$

on $\mathbf{R}^N$ is determined by specifying N functions

$$a_\mu : \mathbf{R}^N \longrightarrow \mathbf{R}.$$

We also denote the vector field by E_a. A parameterized path

$$\gamma : [t^0, t^1] \subset \mathbf{R} \longrightarrow \mathbf{R}^N$$

is called a *trajectory* of the dynamical system

$$\dot{x}(t) = E_a(x(t)) \tag{1}$$

in case it is the unique solution of the initial value problem

$$\dot{x}(t) = E_a(x(t)), \qquad x(t^0) = \gamma(t^0). \tag{2}$$

We define the *vector field/reference point representation* or *VEFREP* of a path η to be the pair (E, R), consisting of a vector field E and a reference or initial point R, where the trajectory is the solution of the initial value problem

$$\dot{x}(t) = E(x(t)), \qquad x(t^0) = R.$$

Note that this representation is not unique. Indeed, several different vector fields could have a given spatial curve as a trajectory, while any point along the spatial curve could serve as the initial value.

We now give an algorithm whose input is a parameterized path

$$\eta : [t^0, t^1] \subset \mathbf{R} \longrightarrow \mathbf{R}^N,$$

and whose output is a labeled, rooted binary tree. We assume for convenience that $t^0 = 0$ and $t^1 = 1$; if not, we can reparameterize. We do not assume that η is a trajectory of the dynamical system (1). To define the tree, we first fix a tolerance $\epsilon > 0$. The tree we define is a subset of the complete rooted binary tree. There are 2^k children at height k from the root: number them left to right from 1 to 2^k. We assign two labels to the jth node v from the left at height k:

$$\kappa(v) = (1/2^k)\left(\eta(\frac{j}{2^k}) - \eta(\frac{j-1}{2^k})\right) \in \mathbf{R}^N$$

and

$$\theta(v) = \eta\left(\frac{j-1}{2^k}\right) \in \mathbf{R}^N.$$

We use the following stopping criterion to grow the tree. If a node has children v and v' with labels κ and κ', respectively, and if $||\kappa - \kappa'|| \leq \epsilon$, then the nodes v and v' are leaves. Here $||\cdot||$ denotes the Euclidean norm. We denote by $T(\eta)$ the tree that arises in this fashion. This tree has a simple interpretation: the θ labels represent points on the path η, while the κ labels represent approximate tangent vectors at those points. The tree is grown until the difference between two adjacent tangent vectors is uniformly small.

Using the tree $T(\eta)$, we now define a vector field $E(\eta)$. The vector field $E(\eta)$ is simply the vector field which interpolates the labels $(\theta(v), \eta(v))$

$$E(\eta)(\theta(v)) = \kappa(v), \tag{3}$$

for all leaves v in $T(\eta)$. Recall that $\theta(v)$ is the point on the curve η corresponding to the node v, and $\kappa(v)$ is the approximate tangent to the curve at that point.

For some applications, it is better to impose an upper bound on the degree of the interpolating functions. Let q denote this bound. In this case, we can define the vector field $E(\eta)$ by requiring that the coefficients b_j^μ minimize the quantity

$$\sum_{\text{leaves } v} ||E(\eta)(\theta(v)) - \kappa(v)||, \tag{4}$$

where the minimum is over vector fields with interpolating functions of degree less than or equal to q.

We conclude this section by defining a specific point $R(\eta)$, associated with a parametrized path

$$\eta : [t^0, t^1] \subset \mathbf{R} \longrightarrow \mathbf{R}^N.$$

Let $T(\eta)$ the corresponding tree and $E(\eta)$ the associated vector field. Consider the trajectory defined by the initial value problem

$$\dot{x}(t) = E_a(x(t)), \qquad x(0) = \eta(t^0).$$

Let γ denote this trajectory. In general, γ is only an approximation to the path η. Define $R(\eta) \in \mathbf{R}^N$ as follows: if the path γ and the unit sphere in $\mathbf{R}^N$ intersect precisely once, let $R(\eta)$ denote this intersection; if they intersect several times, let $R(\eta)$ denote the intersection which occurs first when the list of intersections is ordered in lexigraphical order; otherwise, let $R(\eta)$ denote the closest point between the unit sphere and the trajectory γ.

3 Query Algorithms

Let $\eta \subset \mathbf{R}^N$ denote a path. In this section, we define an index $I(\eta)$ that can be used for storing and accessing the path η. First, fix injective functions

$$h_n : \mathbf{R}^n \longrightarrow \{1, 2, \ldots\},$$

for each $n = 1, 2, \ldots$ Given a path η, first compute its VEFREP $(E(\eta), R(\eta))$ from its rectifying tree $T(\eta)$. Assume the tree $T(\eta)$ has K leaves. Next, view the coefficients of the vector field $E(\eta)$ as a $K \cdot N$ vector, so that the pair $(E(\eta), R(\eta))$ has $(K+1)N$ components. Then the *hash index* $H(\eta)$ associated with η is defined by

$$H(\eta) = h_{(K+1)N}(E(\eta), R(\eta)).$$

We can now assign indices $I(\eta)$ to trajectories sequentially: use the hash index $H(\eta)$ to determine whether the path η has an index assigned to it. If so, use that index; if not, use the next available index.

Suppose that $\gamma_1, \ldots, \gamma_P$ are trajectories of the dynamical system

$$\dot{x}(t) = E_a(x(t)),$$

as a ranges over some parameter space. To each such trajectory γ, let

$$(E(\gamma), R(\gamma))$$

denote its VEFREP representation. Given a parameterized path η, Algorithm 1 below returns the trajectory γ from the database which contains a segment equal to the path η. If there is no such trajectory, null is returned.

Algorithm 1. The input is a parameterized path η, and the output is the trajectory γ from the database answering Query 1. Fix $\epsilon > 0$ and $q > 1$.

Step 1. This step is a precomputation. For each trajectory γ_i, $i = 1, \ldots, P$, compute its VEFREP representation $(E(\gamma_i), R(\gamma_i))$. This depends upon q and ϵ.

Step 2. Given a query path η, compute its rectifying tree $T(\eta)$. This depends upon ϵ. Using $T(\eta)$ and Equation 4, compute its VEFREP representation $(E(\eta), R(\eta))$. This depends upon q.

Step 3. Using the VEFREP $(E(\eta), R(\eta))$, compute the hash index $H(\eta)$. If there is an index in the row $H(\eta)$ of the index table, retrieve the VEFREP representation (E, R) corresponding to this index; otherwise, return null.

Step 4. If Step 3 yielded a VEFREP (E, R), return the trajectory γ which is the solution to the initial value problem

$$\dot{x}(t) = E(x(t)), \qquad x(0) = R;$$

otherwise, return null.

Theorem 3.1 *Assume that the database contains n trajectories. Algorithm 1 answers Query 1 in time* $O(1)$.

Easy modifications of Algorithm 1 can be used to answer Query 2 in time $O(n)$.

4 Conclusion

In this paper, we have described preliminary work concerned with queries of databases containing trajectories of differential equations. Trajectories of differential equations have many different representations. For the types of queries considered here, we have chosen to represent parameterized trajectories

$$\gamma : [t^0, t^1] \subset \mathbf{R} \longrightarrow \mathbf{R}^N$$

by a pair, consisting of a vector field E on $\mathbf{R}^N$ with polynomial coefficients and a point $R \in \mathbf{R}^N$ such that the trajectory is the solution of the initial value problem:

$$\dot{x}(t) = E(x(t)), \qquad x(t^0) = R.$$

We call this a VEFREP representation. Using the VEFREP representation, we have introduced an index $I(\gamma)$ and algorithms to answer queries which retrieve subtrajectories and close by trajectories of a given query trajectory.

References

[1] H. Abelson and G. J. Sussman, Dynamicists' Workbench I: "Automatic Preparation of Numerical Experiments," R. Grossman (editor), *Symbolic Computation: Applications to Scientific Computing*, SIAM, 1989.

[2] R. Grossman, "Querying Databases of Trajectories of Differential Equations I: Data Structures for Trajectories," to appear in *Proceedings of the 23rd Hawaii International Conference on Systems Sciences*, IEEE, 1990.

[3] W. Kim, J. Banerjee, H. T. Chou, J. F. G. Garze, and D. Woelk, "Composite Object Support in an Object Oriented Database System," *Proceedings of OOPSLA 1987.*

[4] W. Kim, K-C Kim, and A. Dale, "Indexing Techniques for Object-Oriented Databases," in *Object-Oriented Concepts, Databases, and Applications*, W. Kim and F. H. Lochovsky, editors, ACM, New York, 1989.

[5] D. Maier and J. Stein, "Indexing in an Object-Oriented DBMS," Proceedings of the 1986 International Workshop on Object-Oriented Database Systems, Pacific Grove, California, 1986.

[6] N. Paton and P. M. D. Gray, "Identification of Database Objects by Key," in *Advances in Object-Oriented Database Systems*, K. R. Dittrick, editor, Springer-Verlag, Berlin, 1988, pp. 280–285.

[7] J. T. Schwartz and M. Sharir, "Identification of Partially Obscured Objects in Two Dimensions by Matching of Noisy *Characteristic Curves*," International J. Robotics Research, 6 (1987), 29–44.

[8] H. Wolfson, "On Curve Matching," *Proceedings of Workshop on Computer Vision, Miami Beach*, IEEE, 1987, 307–310.

A FAST ALGORITHM FOR CONTROL AND ESTIMATION USING A POLYNOMIAL STATE-SPACE STRUCTURE

James R. Shults and Thomas Brubaker
Colorado State University

Gordon K. F. Lee
North Carolina State University

ABSTRACT

One of the major problems associated with the control of flexible structures is the estimation of system states. Since the parameters of the structures are not constant under varying loads and conditions, conventional fixed parameter state-estimators can not be used to effectively estimate the states of the system. One alternative is to use a state-estimator which adapts to the condition of the system.

One such estimator is the Kalman filter. This filter is a time-varying recursive digital filter which is based upon a model of the system being measured. This filter adapts the model according to the output of the system. Previously, the Kalman filter has only been used in an off-line capacity due to the computation time required for implementation. With recent advances in computer technology, it is becoming a viable tool for use in the on-line environment. The following paper describes a distributed Kalman filter implementation for fast estimation of the state of a flexible arm. A key issue, is the sensor structure and initial work on a distributed sensor that could be used with the Kalman filter is presented.

INTRODUCTION

The parameters of flexible structure systems are generally dynamic. They change under varying load and environmental conditions. When there is a need to control such dynamic systems, these parameters must be measured or estimated. These systems are usually very complex and often more parameters are needed for control than can be measured. The parameters which cannot be measured must therefore be estimated in some manner.

With the rapid evolution of computers, the Kalman filter is becoming an excellent tool for estimation of system parameters. Previously, this filter could only be used in off-line applications such as filtering of laboratory data Brubaker. Now, it is becoming useful in on-line environments for state estimation.

The Kalman filter is a time-varying digital filter which is based upon a model of the system being studied. The filter uses signals from the system to adapt the model and estimate system parameters. These parameters along with the measured signals can then be used to control the system. A key issue in the use of parameter estimation is the sensor distribution and use of appropriate sensor types. Along with this is the data fusion issue from sensors to provide appropriate control information. Here we only describe the estimation procedure with comments on the sensor issue. A block diagram of the filter structure is shown in Fig. 1 for a single input system. For multiple sensors, multiple filters could be deployed and data fusion done with filter outputs.

This work was supported by NASA Subcontract G-1926-1.

KALMAN FILTER THEORY

The Kalman filter is a time-varying recursive digital filter which estimates the states of a system from one or more sensor signals. The filter operates on time domain signals using linear least squares estimation that utilizes all of the past data from the output signals. This estimation can provide separation of signal components. These components can be used to determine the states of the system. The Kalman filter can also improve noise reduction on the signals under consideration [Brubaker].

The first step in the design of a Kalman filter is the choice of a linear or linearized signal model that describes the signal or serves as an approximation to the signal. One of the most flexible of these models is a polynomial. The input signal to the filter (output from the system), z(t), is represented by a polynomial of order m. At time t = nT, where T is the sampling period of the system, the state vector for the system is given by

$$z(nT) = \begin{bmatrix} z \\ \frac{dz}{dt} \\ \cdot \\ \cdot \\ \cdot \\ \frac{d^m z}{dt^m} \end{bmatrix}_{t=nT} \tag{1}$$

Here, the system is being represented in canonical state-space form with the components of the state vector being the derivatives of the polynomial model. When a system is to be represented in a different state-space form, a linear transformation can be performed to change the state vector to the desired form.

To use the polynomial model for the Kalman filter, the state vector must be redefined using a Taylor Series representation for each element of z(nT). The resulting state vector is

$$x(nT) = \begin{bmatrix} z \\ T\frac{dz}{dt} \\ \cdot \\ \cdot \\ \cdot \\ \frac{T^m}{m!}\frac{dz^m}{dt^m} \end{bmatrix}_{t=nT} \tag{2}$$

The state of the system at t = (n+h)T can now be described in terms of the state at t=nT by the relationship

$$x[(n+h)T] = \Phi(h)\, x[nT] \tag{3}$$

where $\Phi[h]$ is the following (m+1) X (m+1) state transition matrix:

$$\Phi[h] = \begin{bmatrix} 1 & h & .. & h^m \\ 0 & 1 & .. & mh^{m-1} \\ . & . & .. & . \\ . & . & .. & . \\ 0 & 0 & .. & 1 \end{bmatrix} \tag{4}$$

The filter is implemented by first producing forecasts of the estimate

$$x_1(nT) = \Phi(1)x[(n-1)T] \tag{5}$$

and covariance matrix

$$S_1(nT) = \Phi(1)S[(n-1)T]\Phi^T(1) + Q$$

at t = nT using the previous estimate, x[(n-1)T], and covariance matrix, S[(n-1)T]. These forecasts are obtained by using the state transition relationship given in (4). The matrix Q is the covariance of the driving noise. It allows the designer to "fade" the effects of past inputs. The covariance forecast is then used to calculate the Kalman Gain Matrix,

$$K(nT) = S_1(nT)M^T[\sigma^2 + MS_1(nT)M^T]^{-1} \tag{7}$$

The term σ^2 is the variance of the measurement noise. The matrix M is a row matrix which relates the measurable state variables to the actual measurements. For this filter, the measurements are assumed to be components of the state-vector z(Nt) given in (1). Therefore, M relates the estimate vector x(Nt) given in (2) to the measured components of z(Nt). The Kalman Gain Matrix is then used to obtain the covariance estimate,

$$S(nT) = [I - K(nT)M]S_1(nT) \tag{8}$$

and the state-vector estimate,

$$x(nT) = x_1(nT) + K(nT)[y(nT) - Mx_1(nT)] \tag{9}$$

In (9), the term y(nT) is the data measurement vector at t=Nt which consists of the measured components of z(nT). Equations (1) through (9) are the basis for the Kalman filter. A complete derivation for these equations can be found in many texts for example, Liebelt and Meditch.

KALMAN FILTER DESIGN

With the choice of a polynomial model for the input signal, the design of a Kalman filter involves the determination of a few key parameters. These parameters are set according to the system and design specifications.

The first group of parameters which must be determined are associated directly with the properties of the system. The first is the sampling period, T. This period is usually set according to the nyquist rate,

$$f_s = \frac{1}{T} \geq 2f_m \tag{10}$$

where f_m is the maximum frequency of the bandlimited input signal and f_s is the sampling frequency.

The second parameter is the variance of the sensor noise, σ^2. This parameter is a property only of the sensors which are employed. Another term which is determined from the sensors which are utilized is the Data Measurement Vector. This is the number of terms in the state vector of equation 1 which can be directly measured.

The remaining filter parameters are determined using both design and system specifications. The order of the polynomial model must be set according to the sampling period and the angular velocity of the input signal oscillations. The calculation time of the filter is larger for higher orders. The order must be small enough to allow the calculation time to be smaller than the sampling period. On the other hand, the order must be large enough to allow the filter to track the input signal well.

The key design parameter of the Kalman filter is the covariance matrix of the driving noise, Q. For this paper, the driving noise is assumed to be uncorrelated white noise. This simplifies the matrix Q to a diagonal matrix. After testing different forms for the Q matrix, little difference was found. Therefore, the matrix Q was taken to be the identity matrix times a constant, f.

The constant f is known as the fading factor. This term determines how the filter will handle past data. When f = 0, the filter is simply an expanding memory filter. All past data is used evenly to calculate the present estimate. This will cause the variance of the estimate and its covariance matrix to decrease to zero as time increases, but deterministic errors will become large. If the fading factor is greater than zero, more emphasis is placed on the present sample than on past samples. Thus, the past samples are faded from the filter's memory. Larger values of f cause past samples to fade more quickly. The fading of past samples causes the Kalman filter to have a much smaller deterministic error, but the variance of the estimate and its covariance approach the values at the input.

One way to minimize the deterministic error associated with a small fading factor is to periodically reinitialize the Kalman filter. This requires establishing a relationship between the frequency of initialization and the fading factor. The fading factor should be the minimum value for which the deterministic error just prior to reinitialization is within specifications. This value will cause the Kalman filter to provide maximum noise reduction while meeting deterministic error specifications.

To reinitialize the Kalman filter, a nonrecursive filter is used. This filter utilizes a window of the past samples to estimate the state vector and its covariance. These parameters are then passed to the Kalman filter.

DESIGN EXAMPLE

The Kalman filter was tested in two separate operating conditions. The first was with data acquired from the hub of the flexible arm at CSU. The second test was with an eighth order model of the flexible arm. This test included the simulation of the system with control in a closed-loop environment. A program has been developed which allows the design and testing of these and other filter structures. The results of these experiments are discussed in the following paragraphs.

The data acquired from the hub of the flexible arm was position information sampled at a rate of 100 samples per second. A polynomial model of order 3 was required to accurately represent the position of the hub. Since velocity information was not available, a Data Measurement Vector of one was used. The variance of the sensor noise was arbitrarily set at 10^{-4}. The filter was reinitialized every 100 samples, 1 second, and a fading factor of 10^{-9} was used.

The outputs of the filter are shown in Figs. 2 through 5. Figure 2 demonstrates the ability of the filter to estimate the position signal. Figure 3 displays the estimate of the velocity. Figures 4 and 5 show the estimates of the second and third derivatives of the position signal. No actual data was available for the second and third derivatives. Therefore, the accuracy of these estimates could not be determined.

The second test of the filter involved an eighth order simulation of the flexible arm with the filter outputs used as control feedback. The simulation provided the position of the tip as input to the filter. A third order polynomial model was used for the Kalman filter. The variance of the sensor noise was taken to be 10^{-4} and the fading factor was $5x10^{-12}$. The feedback control was implemented as simple proportional position only negative feedback. The feedback gain was set to 2.5

The results of this simulation are shown in Figs. 6 through 9. Figure 6 shows a comparison of the position signal before and after the Kalman filter. Figures 7 through 9 show the first, second and third derivatives of position.

FUTURE WORK

Future work on the Kalman filter consists of implementation issues. A good design method has been set forward, but many implementation issues have not been addressed. The major implementation problem comes with the selection of hardware to run the Kalman filter. This hardware must consist of a microprocessor which is fast enough to meet the sampling rate, but inexpensive enough to make its use feasible. Another major problem comes in the sensors which will provide state information for the Kalman filter. These sensors must be selected for each application such that they provide accurate information with a low noise level. A final problem which must be addressed is a method to download the Kalman filter program to the hardware. This downloading could be done by the design program with the appropriate interfaces. After these problems have been resolved, the Kalman filter will provide a very effective state estimator in many applications.

A NEW SENSOR

Within the past few months our group has designed and performed initial tests on a distributed fiber optic sensor. Here, the fiber is connected to a flexible structure over a one meter length. The fiber is excited with a milliwatt laser and the defraction pattern out of the end is used to provide an estimate of displacement. Figures 10 and 11 illustrate the results via change in the output pattern. For implementation a CCD memory could be used to store the pattern and subsequently the information is driven into a computer where basic pattern recognition techniques are used to generate good estimates of displacement. A well designed system can also be used to estimate velocity. Within the context of this paper, a two-dimensional Kalman filter can be used to estimate parameters. Note that in Figs. 10 and 11, black and white images are shown. In a physical system color will be used.

CONCLUSIONS

An investigation has been started into the usefulness of the Kalman filter as a state estimator in an on-line environment. Previously, the filter has been used strictly in an off-line capacity to do data analysis. With the advances in computing speed, the filter is now becoming feasible as a real-time state estimator.

A Kalman filter based on a polynomial state-space model has been tested on flexible structure data. This filter has proven to give excellent state-estimation and noise reduction in such systems. The polynomial model provides for a very easy design of the filter. Do to the ease of design, a program has been written to provide assistance in the design process.

The Kalman filter design program provides a very straight forward design methodology with an interactive graphics approach. This approach allows the designer to see how well the filter works and the effects of changes in the design parameters. When the program is combined with the appropriate hardware, a very effective state estimation tool will become available for use in the real-time environment.

REFERENCES

1. Brubaker, T. A. et. al. Linear Digital Filtering for Laboratory Automation, Proceedings of the IEEE, Vol. 63, No. 10, October 1975.

2. Liebelt, P. B., An Introduction to Optimal Estimation, Addison-Wesley 1967.

3. Medich, J. S., Stochastic Optimal Linear Estimation and Control, McGraw-Hill 1969.

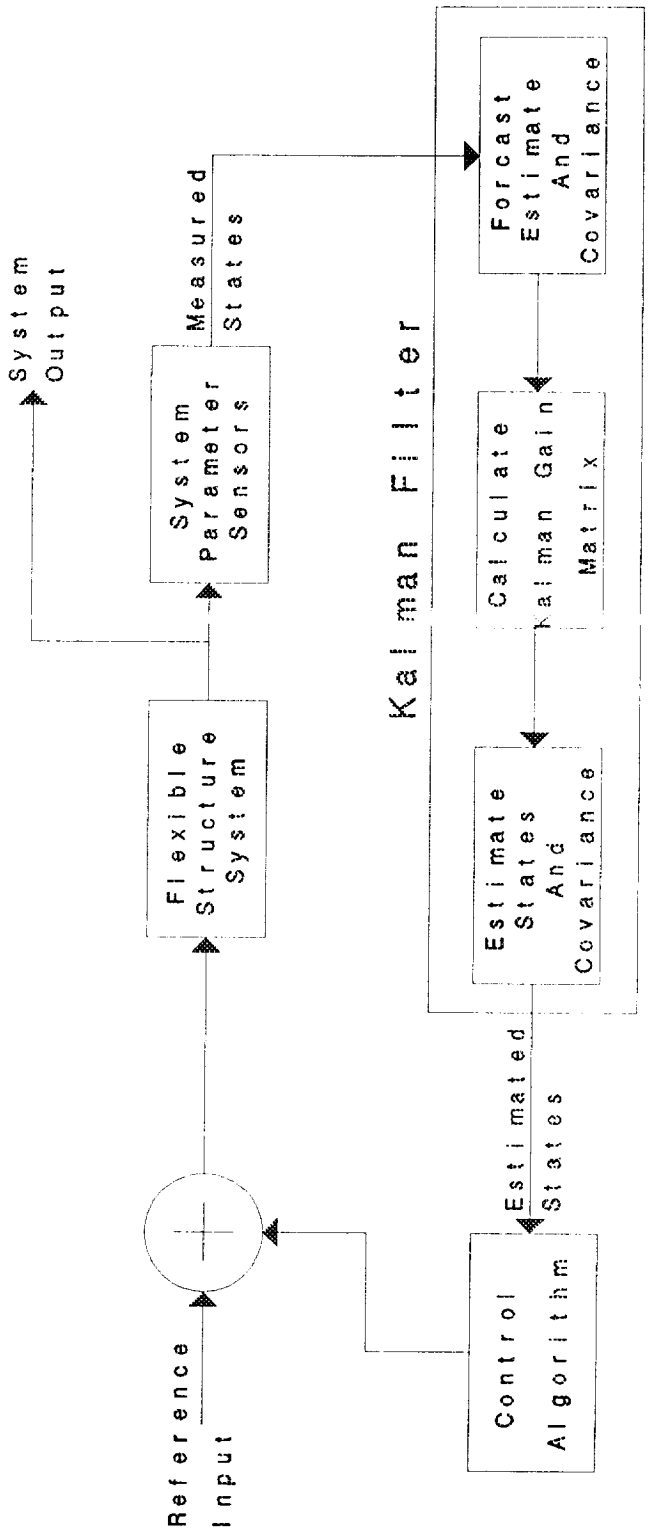


Figure 1. Block Diagram of Kalman Filter System.
Note: For multiple sensors, multiple distributed Kalman Filters could be employed with data fusion a key, important and still a research issue.

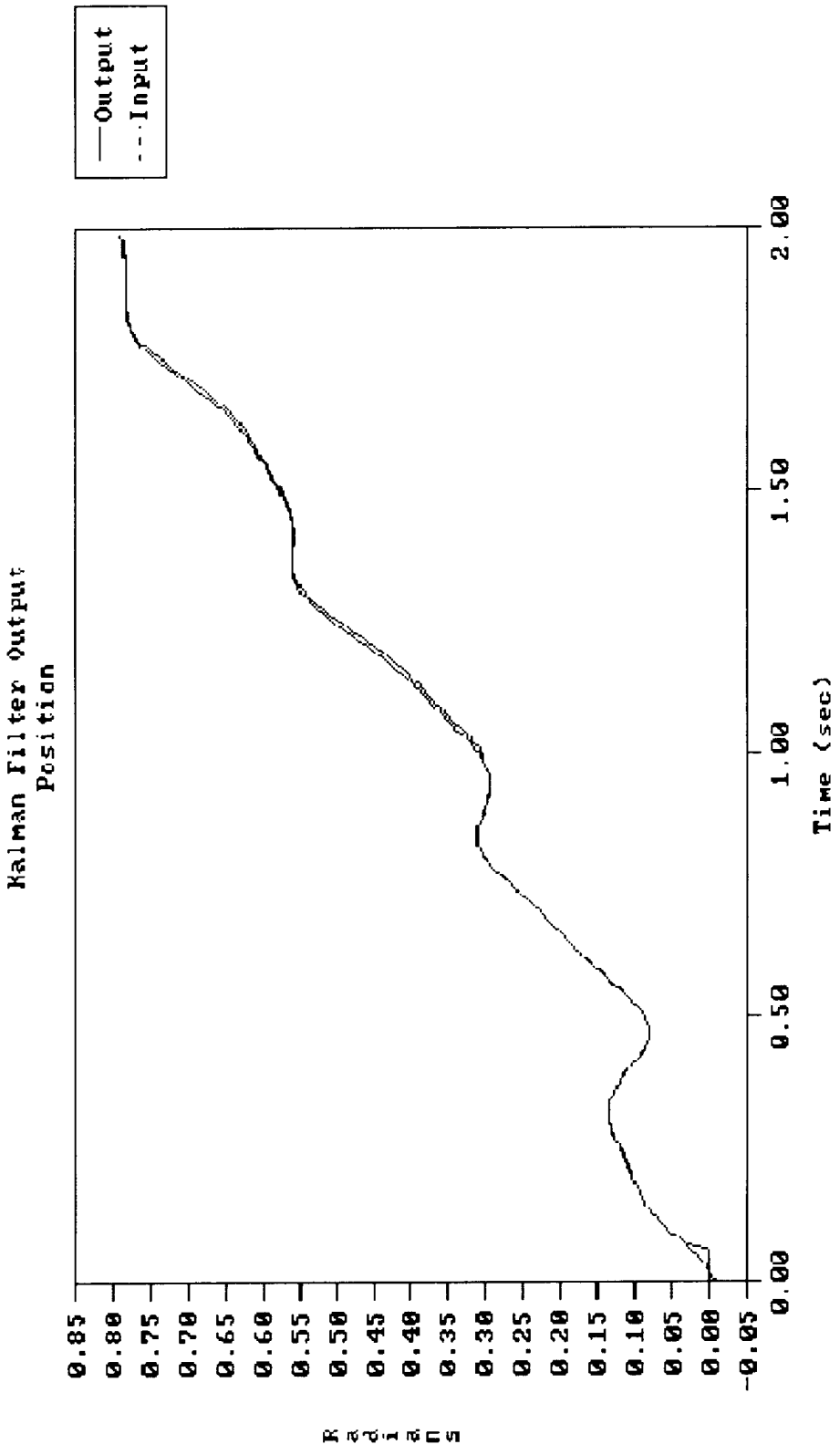


Figure 2. Position Output of Kalman Filter.

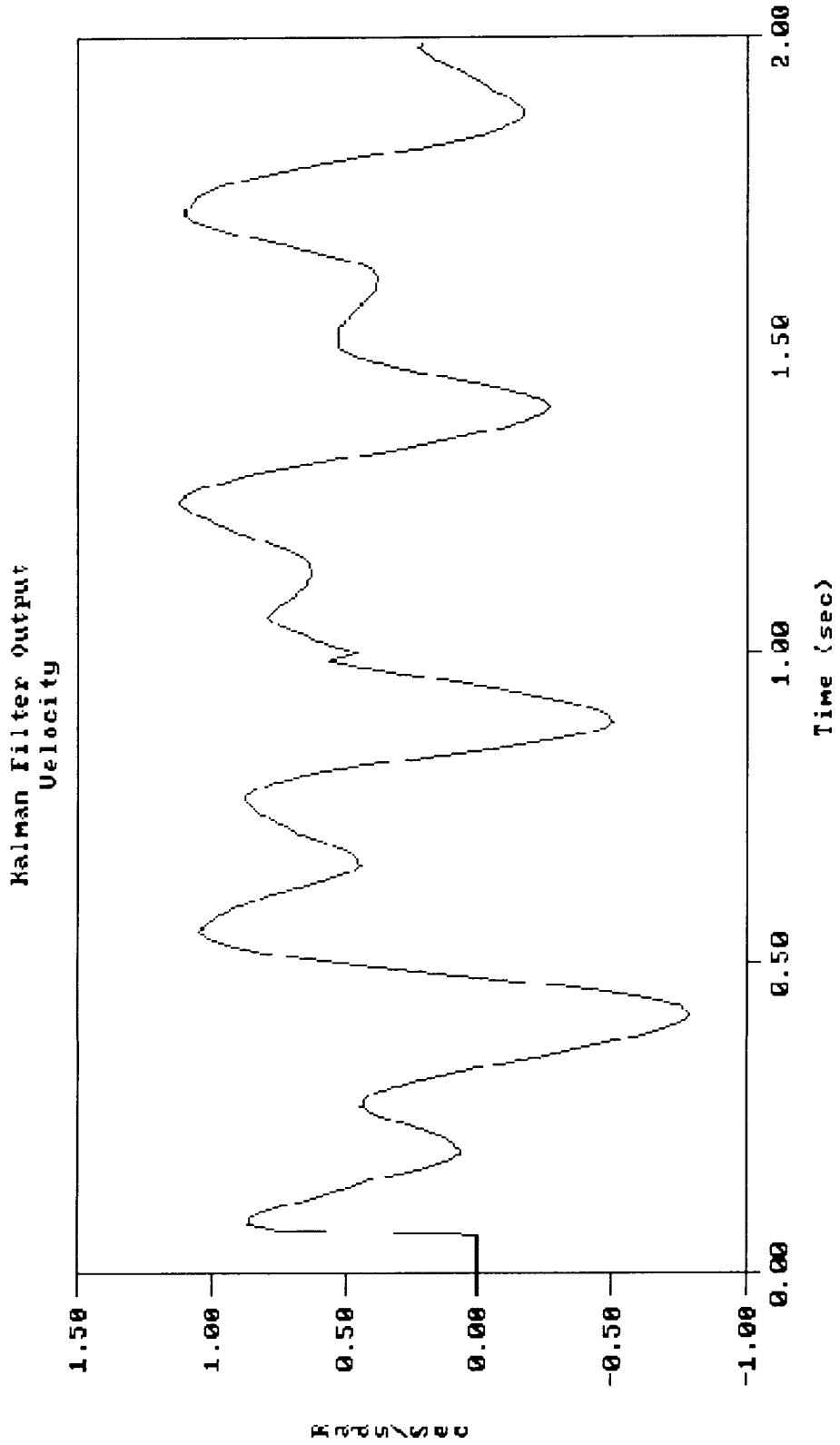


Figure 3. Velocity Output of Kalman Filter.

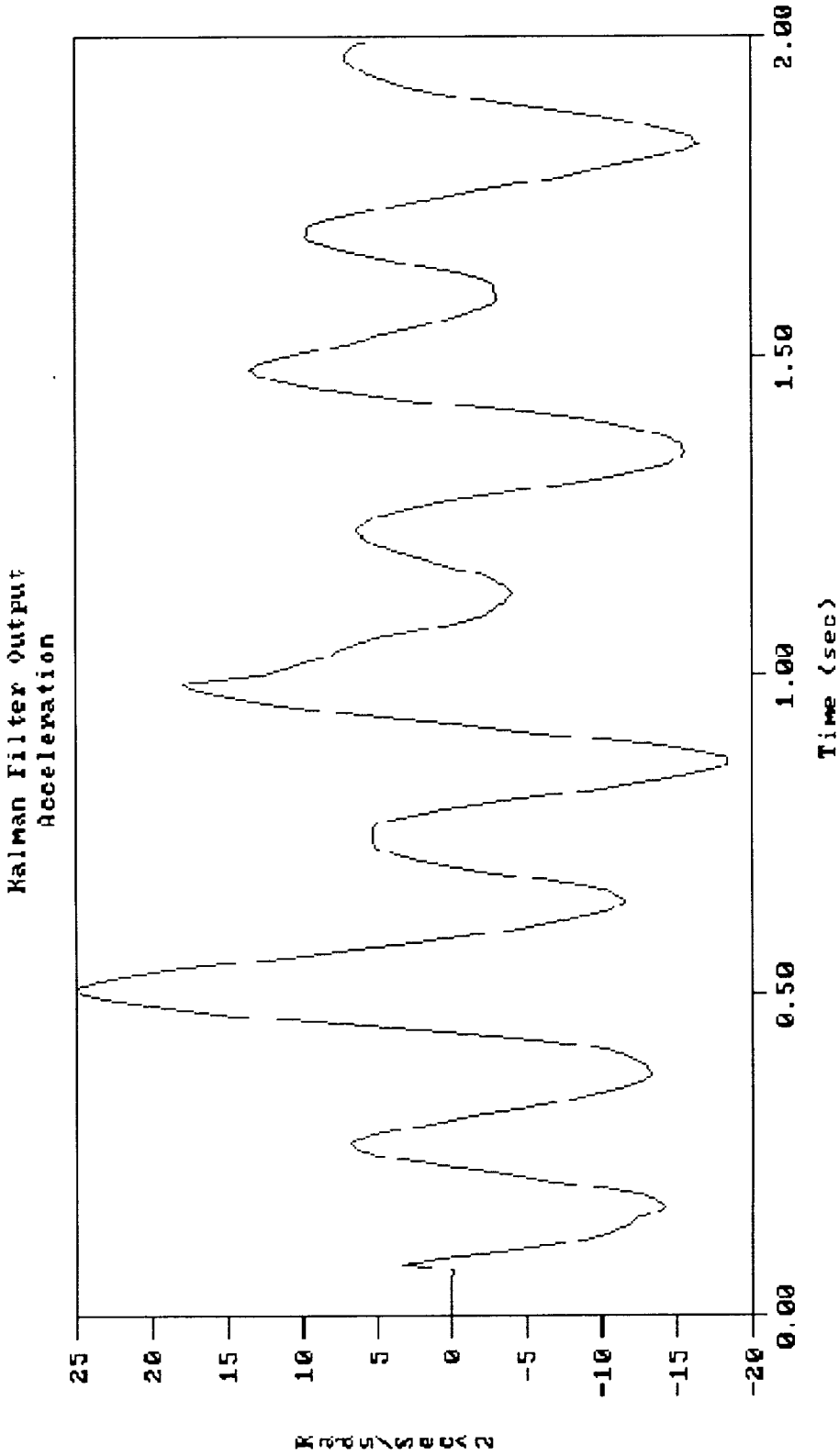


Figure 4. Acceleration Output of Kalman Filter.

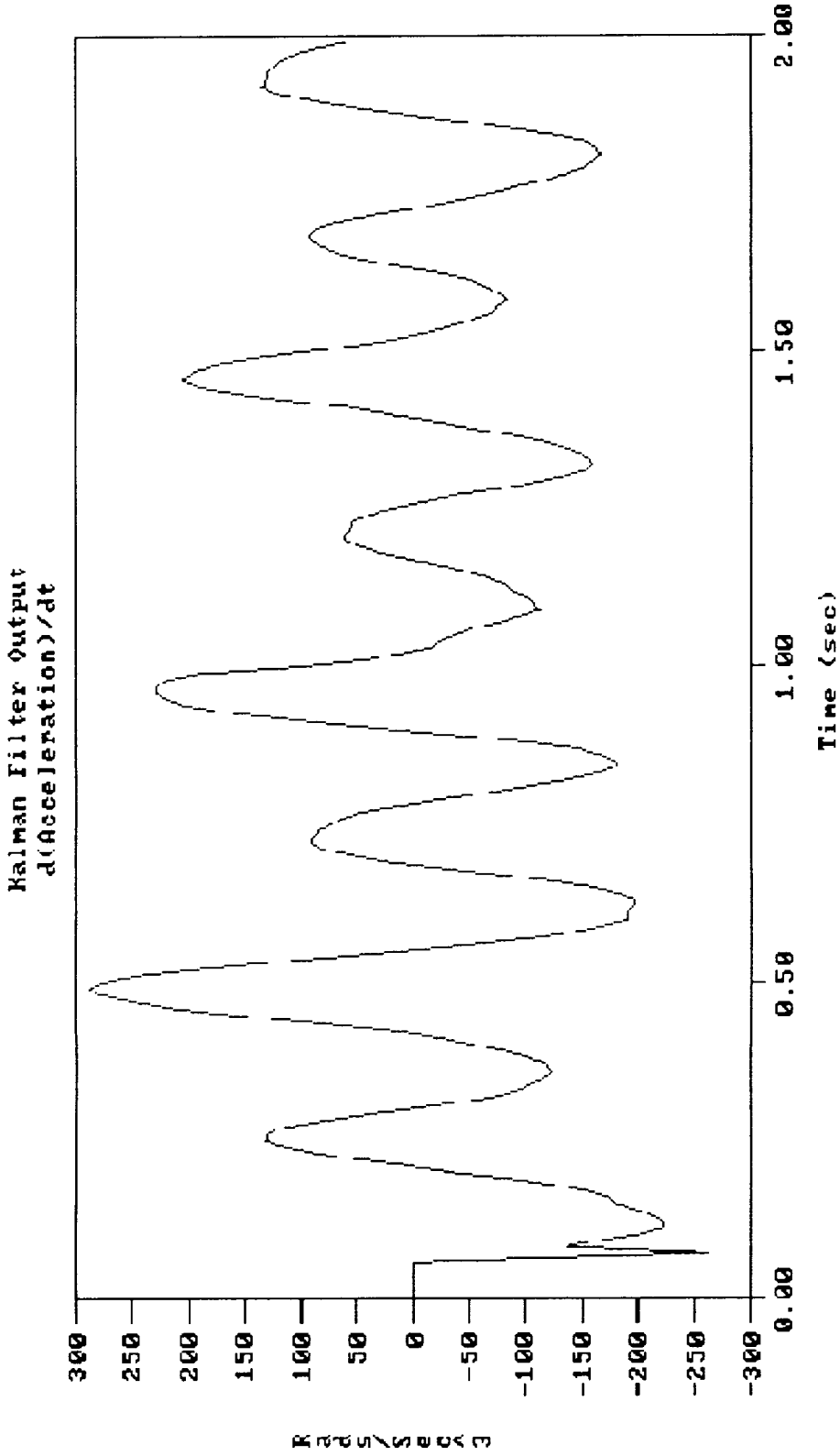


Figure 5. $d^3(\text{Position})/dx^3$ Output of Kalman Filter.

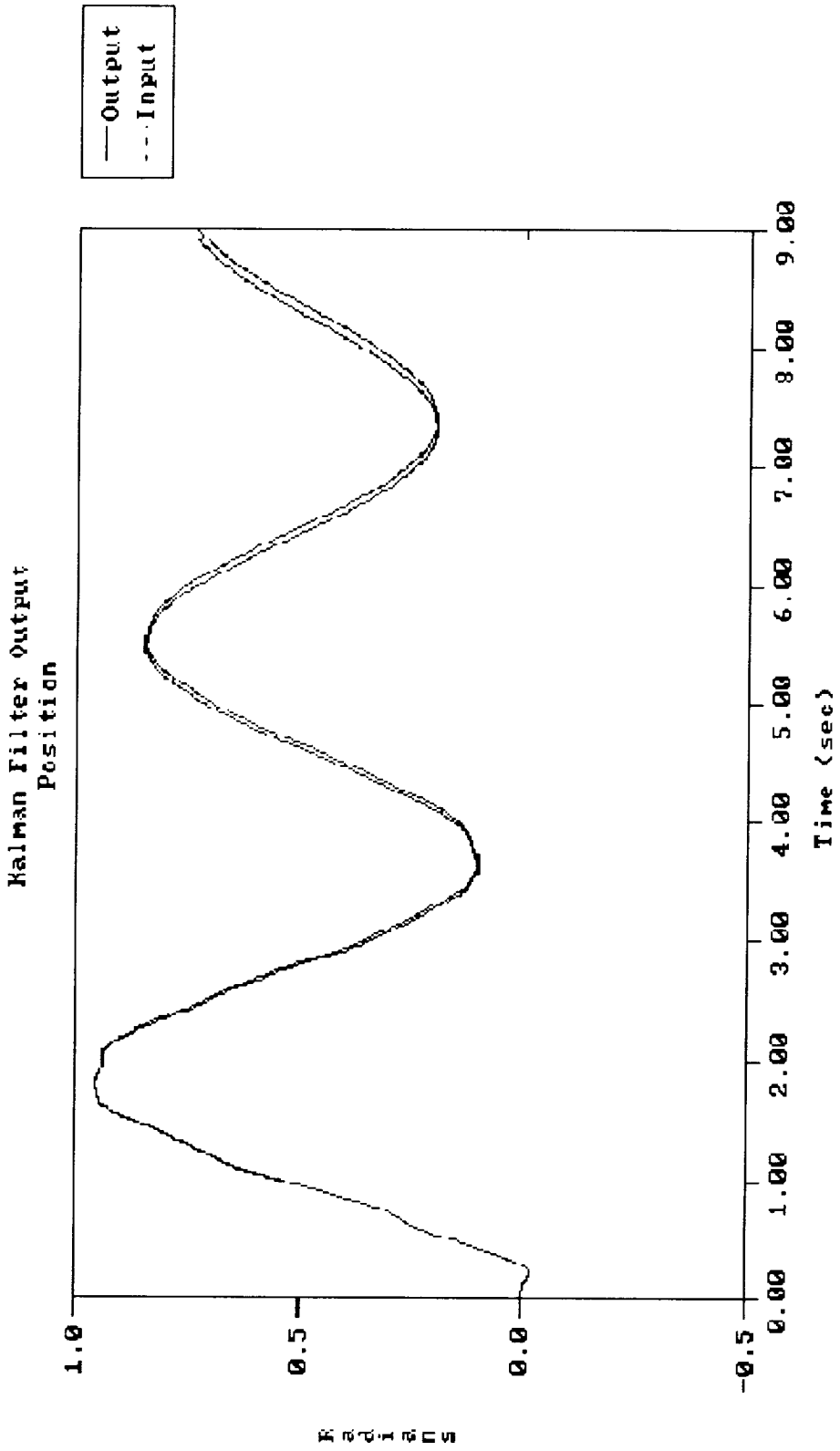


Figure 6. Position Output of Kalman Filter with Flexible Arm Simulation.

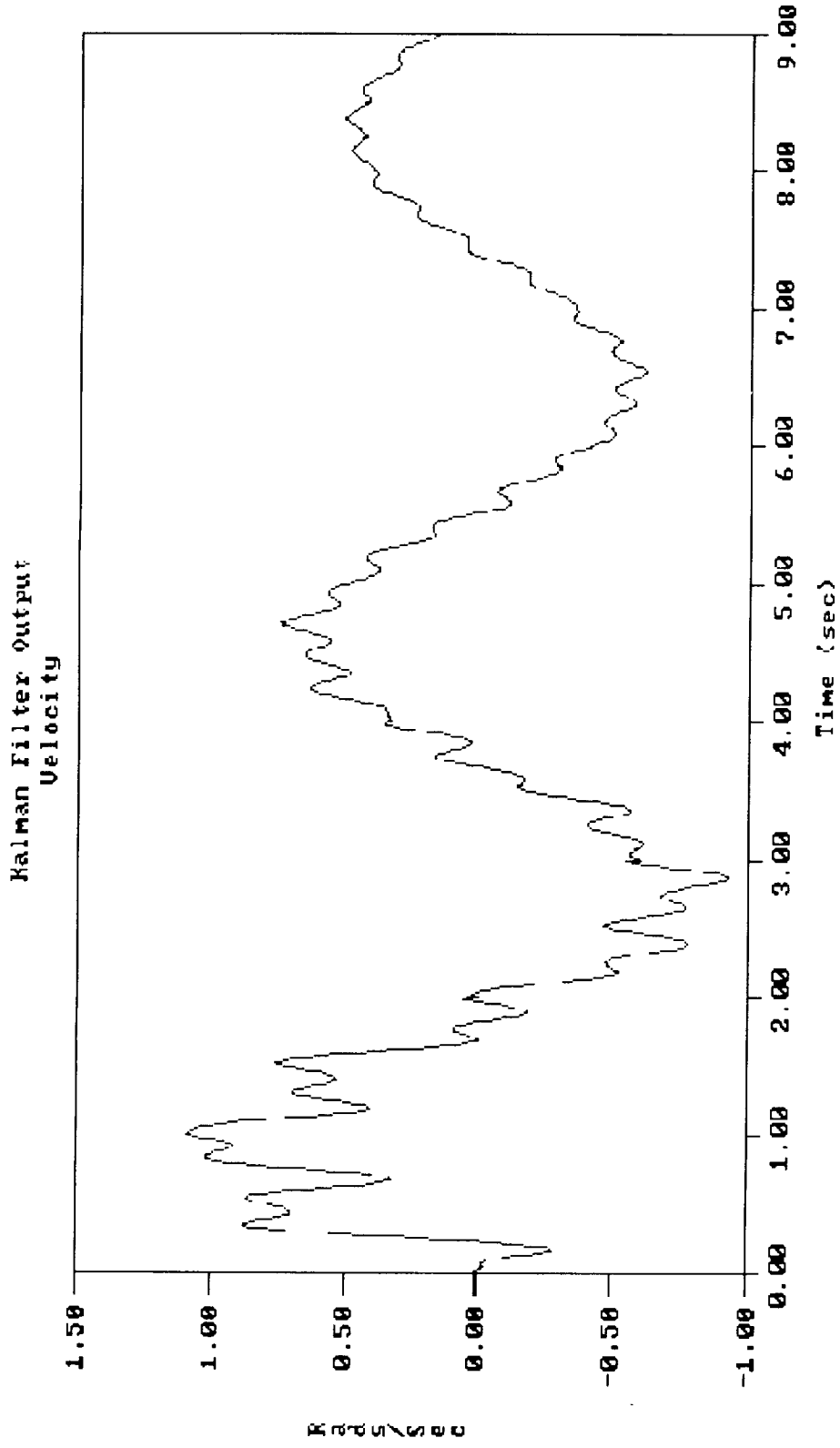


Figure 7. Velocity Output of Kalman Filter with Flexible Arm Simulation.

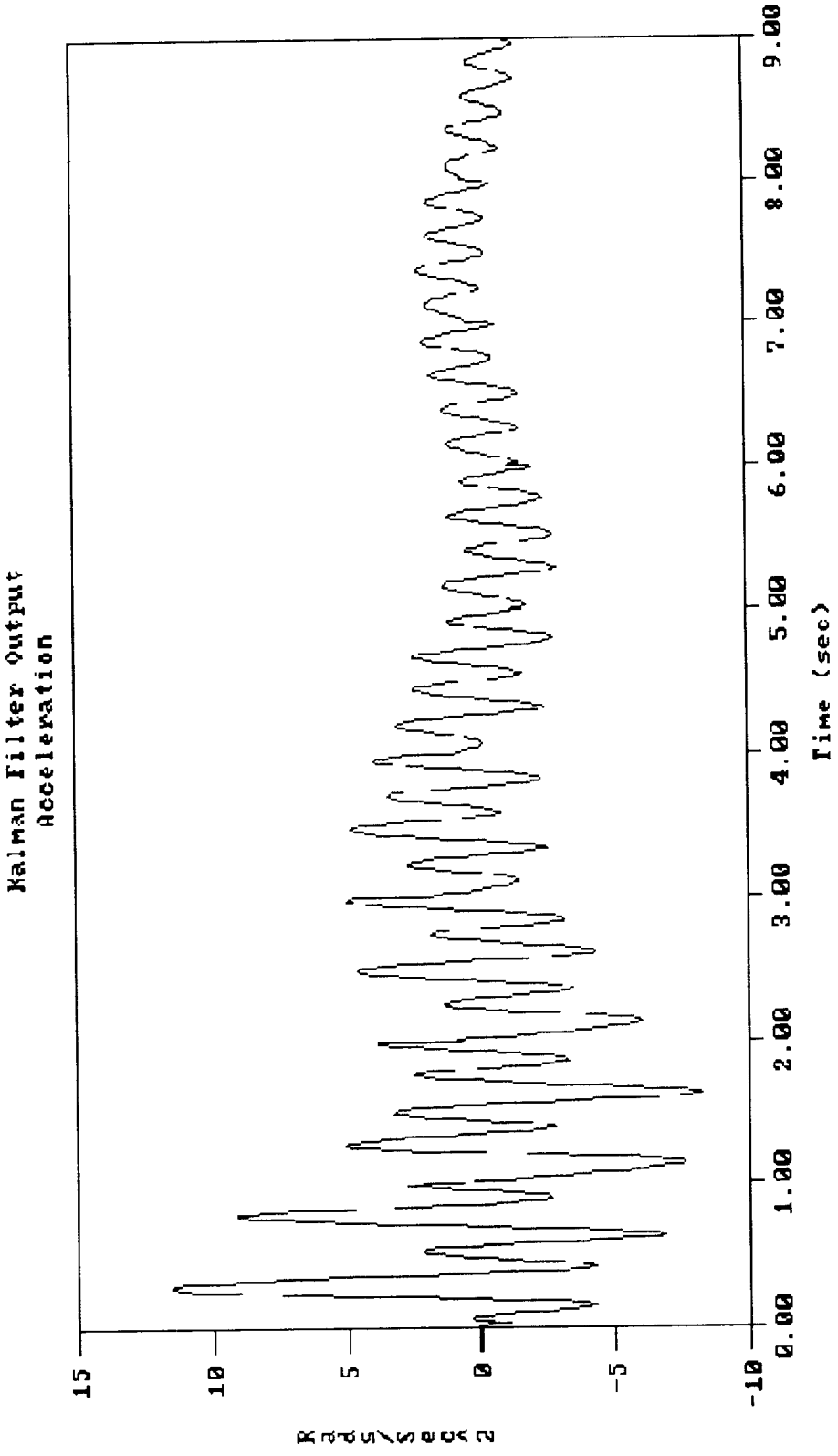


Figure 8. Acceleration Output of Kalman Filter with Flexible Arm Simulation.

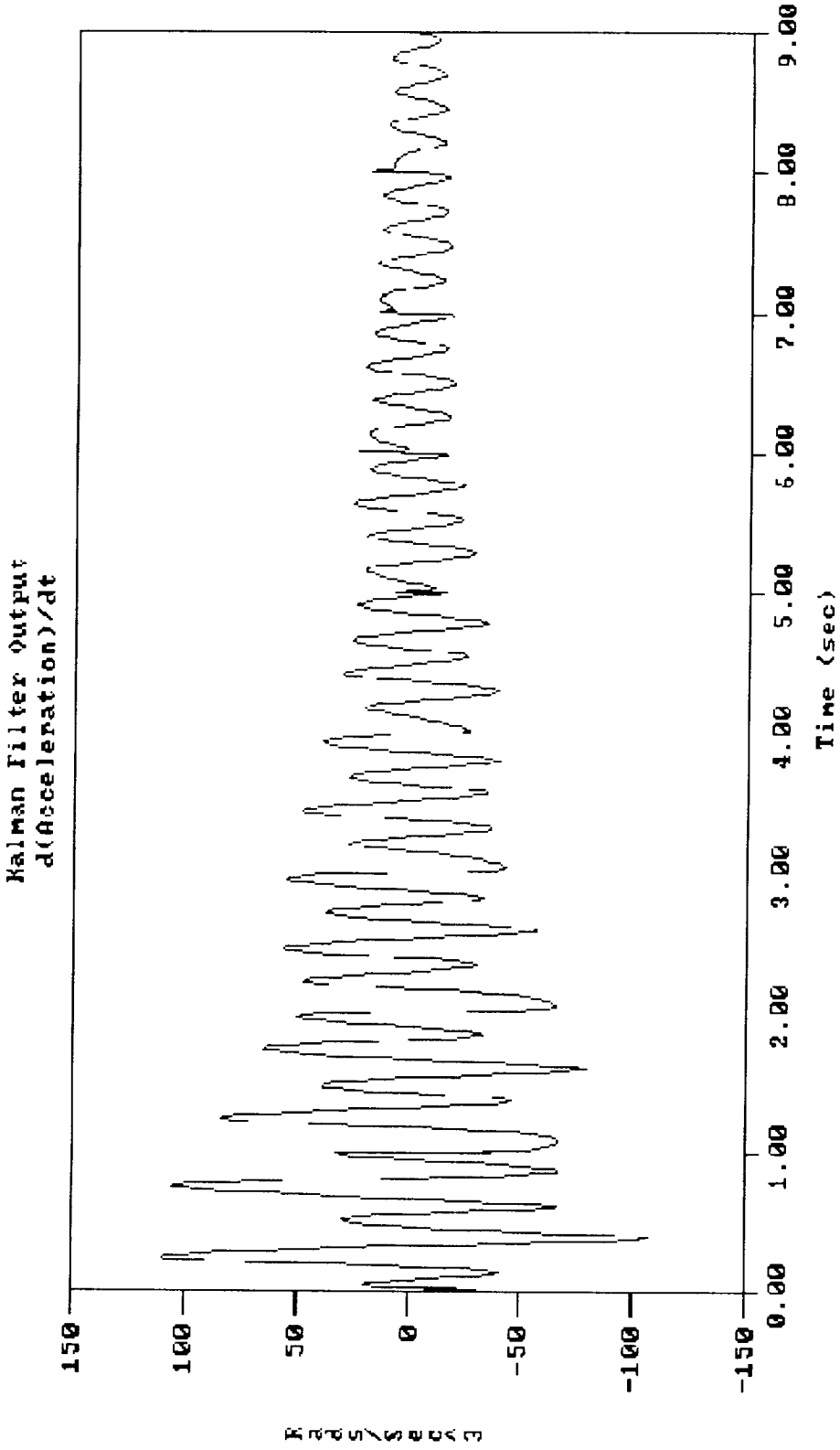


Figure 9. $d^3(\text{Position})/dx^3$ Output of Kalman Filter with Flexible Arm Simulation.

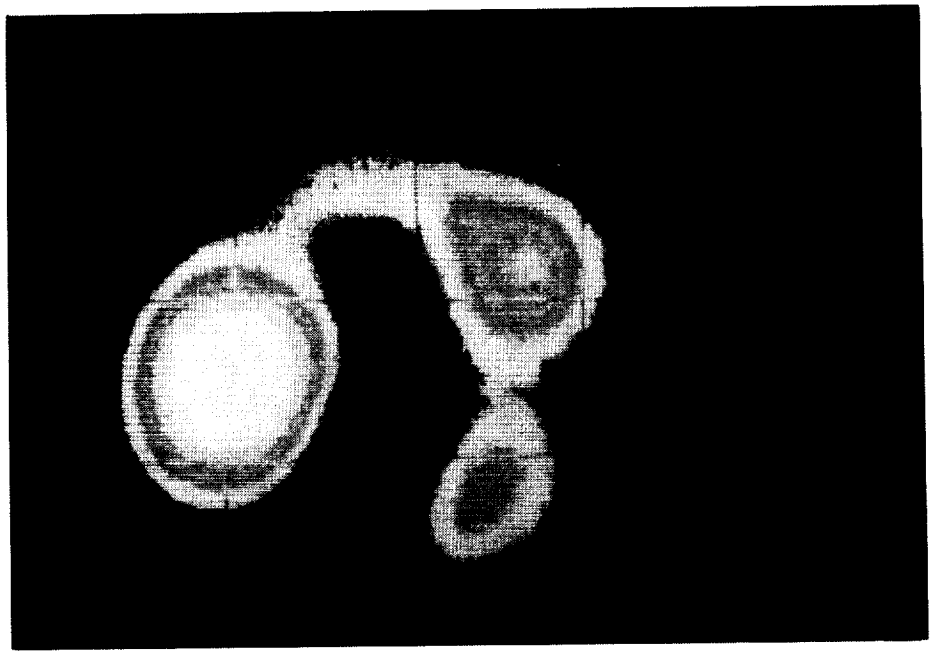


Figure 10. Fiber-Optic Output With Zero Beam Displacements

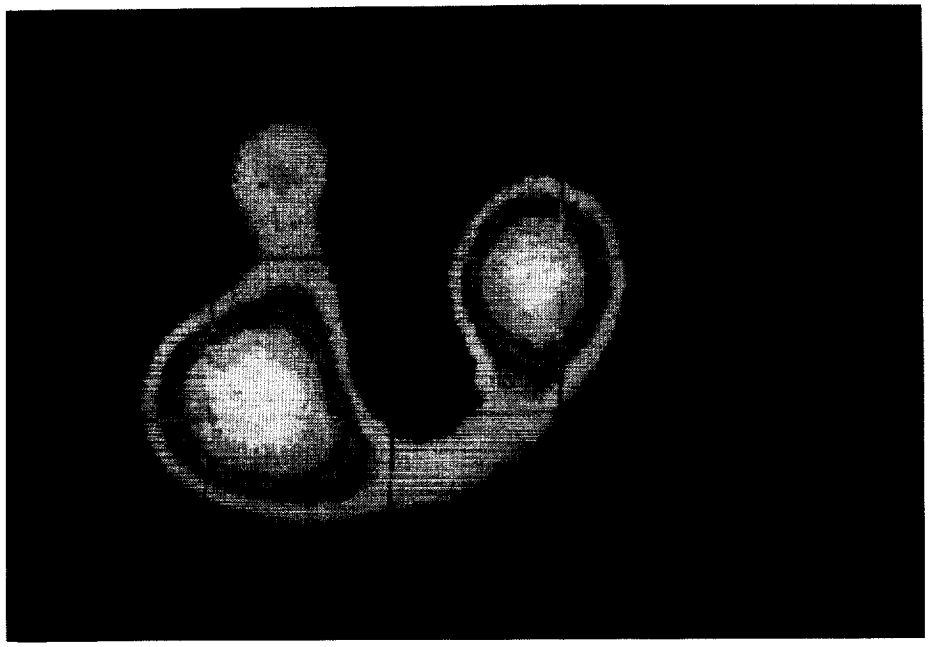


Figure 11. Fiber Optic Output With 5 cm Displacement

SUPERCOMPUTER OPTIMIZATIONS FOR STOCHASTIC OPTIMAL CONTROL APPLICATIONS*

Siu-Leung Chung, Floyd B. Hanson and Huihuang Xu

Laboratory for Advanced Computing
Department of Mathematics, Statistics, and Computer Science
University of Illinois at Chicago
P. O. Box 4348; M/C 249
Chicago, IL 60680

SUMMARY

Supercomputer optimizations for a computational method of solving stochastic, multibody, dynamic programming problems are presented. The computational method is valid for a general class of optimal control problems that are nonlinear, multibody dynamical systems, perturbed by general Markov noise in continuous time, i.e., nonsmooth Gaussian as well as jump Poisson random white noise. Optimization techniques for vector multiprocessors or vectorizing supercomputers include advanced data structures, loop restructuring, loop collapsing, blocking, and compiler directives. These advanced computing techniques and supercomputing hardware help alleviate Bellman's *curse of dimensionality* in dynamic programming computations, by permitting the solution of larger multibody problems. Possible applications include lumped flight dynamics models for uncertain environments, such as large scale and background random aerospace fluctuations.

INTRODUCTION

The objective of this research is to provide a general, highly optimized, computational treatment of stochastic optimal control applications in continuous time. Advanced computing techniques have been implemented so that stochastic dynamic programming algorithms can be used to solve larger optimal control problems than possible by ordinary computing methods. Optimization techniques will help alleviate *Bellman's curse of dimensionality*, in that the computational and memory requirements grow exponentially as the dimension of the state space increases, limiting the size of the control problem that can be computed. Computer optimization techniques can help alleviate Bellman's curse by permitting larger, but still hardware limited problems to be computed. Optimization consists of parallelization and vectorization methods to enhance performance on advanced computers, such as parallel processors and vectorizing supercomputers. Preliminary results for massively parallel processors are also presented.

General Markov random noise in continuous time consists of two kinds, Gaussian and Poisson. Gaussian white noise, being continuous but nonsmooth, is useful for modeling background random fluctuations, such as turbulence and moderate environmental variations. Poisson white noise (its frequency spectrum is also flat like Gaussian noise), being discontinuous, is useful for modeling large random fluctuations, such as shocks, collisions, unexpected external events and large environmental changes. Our general feedback control approach combines the treatment of both linear and nonlinear (i.e., singular and

*This work was supported by the National Science Foundation Computational Mathematics Program under grant DMS-88-06099 at the University of Illinois at Chicago, by the Argonne National Laboratory Advanced Computing Research Facility, by the University of Illinois at Urbana National Center for Supercomputing Applications, and by the UIC Workshop Program on Scientific Supercomputing.

nonsingular) control through the use of small to moderate quadratic costs. The methods also handle deterministic and stochastic control in the same code, making it convenient for checking the effects of stochasticity on the application. Some actual applications are models of resources in an uncertain environments [16], [13], [8].

The Markov, multibody dynamical system is illustrated in Figure 1 and is governed by the stochastic differential equation (SDE):

$$d\mathbf{y}(s) = \mathbf{F}(\mathbf{y}, s, \mathbf{u})ds + G(\mathbf{y}, s)d\mathbf{W}(s) + H(\mathbf{y}, s)d\mathbf{P}(s) \ , \tag{1}$$

with initial value $\mathbf{y}(t) = \boldsymbol{x}$, $0 < t < s < t_f$, $\mathbf{y}(s) \in \mathcal{D}_y$, $\mathbf{u} \in \mathcal{D}_u$, where $\mathbf{y}(s)$ is the $m \times 1$ multibody state vector at time s starting at time t, $\mathbf{u} = \mathbf{u}(\mathbf{y}, s)$ is the $n \times 1$ feedback control vector, $\mathbf{F}$ is the $m \times 1$ deterministic nonlinearity vector, $\mathbf{W}$ is the r-dimensional normalized Gaussian white noise vector, $\mathbf{P}$ is the independent q-dimensional Poisson white noise vector with jump rate vector $[\lambda_i]_{q\times 1}$, G is an $m \times r$ diffusion coefficient array, and H is an $m \times q$ Poisson amplitude coefficient array. In a more general treatment, the Poisson jump amplitude can also be random.

The control criterion is the optimal expected cost performance,

$$V^*(\mathbf{x}, t) \ = \ \min_{\mathbf{u}} \ [\underset{\{\mathbf{P},\mathbf{W}\}}{\text{MEAN}} \ [V[\mathbf{y}, s, \mathbf{u}, \mathbf{P}, \mathbf{W}] \mid \mathbf{y}(t) \ = \ \mathbf{x}]] \ , \tag{2}$$

where the random total cost is

$$V[\mathbf{y}, t, \mathbf{u}, \mathbf{P}, \mathbf{W}] \ = \ \int_t^{t_f} ds \ C(\mathbf{y}(s), s, \mathbf{u}(\mathbf{y}(s), s)) \ , \tag{3}$$

on the time horizon $(t, \ t_f)$. The instantaneous cost function $C \ = \ C(\mathbf{x}, t, \mathbf{u})$ is assumed to be at least a quadratic function of the control,

$$C(\mathbf{x}, t, \mathbf{u}) \ = \ C_0(\mathbf{x}, t) \ + \ \mathbf{C}_1^T(\mathbf{x}, t)\mathbf{u} \ + \ \tfrac{1}{2}\mathbf{u}^T C_2(\mathbf{x}, t)\mathbf{u} \ . \tag{4}$$

C_2 is assumed to be positive definite, so that large controls are much more costly on a per unit basis. In addition, the dynamics in (1) are assumed to be linear in the controls,

$$\mathbf{F}(\mathbf{x}, t, \mathbf{u}) \ = \ \mathbf{F}_0(\mathbf{x}, t) \ + \ F_1(\mathbf{x}, t)\mathbf{u} \ , \tag{5}$$

remaining nonlinear in the state variable $\mathbf{x}$.

The Bellman functional PDE of dynamic programming (or Hamilton-Jacobi-Bellman Equation),

$$\begin{aligned} 0 \ &= \ \frac{\partial V^*}{\partial t} \ + \ \mathcal{H}[V^*] \\ &\equiv \ \frac{\partial V^*}{\partial t} \ + \ \mathbf{F}_0^T \nabla V^* \ + \ \tfrac{1}{2} G G^T(\mathbf{x}, t) : \nabla \nabla^T V^* \\ &+ \ \sum_{l=1}^{q} \lambda_l \cdot [\ V^*(\mathbf{x} + \mathbf{H}_l(\mathbf{x}, t), t) - V^*(\mathbf{x}, t) \ ] \\ &+ \ C_0 \ + \ (\tfrac{1}{2}\mathbf{U}^* - \mathbf{U}_R)^T C_2 \mathbf{U}^* \ , \end{aligned} \tag{6}$$

follows from the generalized *Itô* chain rule for Markov SDEs as in [7] and [16]. Here, $\mathbf{U}^*$ is the optimal feedback control computed by constraining the unconstrained or regular control,

$$\mathbf{U}_R(\mathbf{x},t) \;=\; -C_2^{-1}(\mathbf{C}_1 \;+\; F_1\nabla V^*)\;, \tag{7}$$

to the control set D_u, under the assumption of positive definite quadratic costs. In general, the Bellman equation (6) is nonlinear with discontinuous coefficients due to the quadratic last term, $(\frac{1}{2}\mathbf{U}^* - \mathbf{U}_R)^T C_2 \mathbf{U}^*$, in (6) and due to the compact relationship between the constrained, optimal control and the unconstrained, regular control,

$$U_i^*(\mathbf{x},t) = \min[U_{\max,i}, \max[U_{\min,i}, U_{R,i}(\mathbf{x},t)]], \tag{8}$$

for $i = 1$ to n controls. Here, $\mathbf{U}_{\min}$ is the minimum control constraint vector and $\mathbf{U}_{\max}$ is the maximum. As the constraint components are attained, the optimal control component U_i^*, changes from the regular control component, $U_{R,i}$, to components of the constraints, $U_{min,i}$ or $U_{max,i}$, which in general are functions of state and time. In (6), the symbol (:) denotes the scalar matrix product $A : B = \sum_{i=1}^{m}\sum_{j=1}^{m} A_{ij}B_{ij}$, assuming B is symmetric. It is important to note that the principal equation, the Bellman equation (6), is an exact equation for the optimal expected value V^* and does not involve any sampling approximations such as the use of random number generators in simulations.

As the number of state variables, m, increases, the spatial dimension rises, and computational difficulties are present that can compare to those of three-dimensional fluid dynamics computations. Thus there is a great need to make use of advanced-architecture computers, to use parallelization as well as vectorization, in order to solve larger state space systems. The Panel on Future Directions in Control Theory [6] stresses the importance of making gains in such areas as nonlinear control, stochastic control, optimal feedback control and computational methods for control. This paper is a report on our efforts to treat all of the above mentioned areas combined from the point of view of computational control.

SYMBOLS

$\mathbf{C}, C_0, C_1, C_2$	cost coefficients (eq. (4))
$\mathbf{DX}, DT$	state mesh increment, time increment (eq. (9))
$\mathbf{F}, \mathbf{F}_0, F_1, FV$	nonlinearity function coefficients (eq. (5))
G	Gaussian noise amplitude matrix (eq. (1))
$H, \mathbf{H}_l$	Poisson noise jump amplitude (eq. (1), (6))
$\mathcal{H}$	Hamiltonian for Bellman Equation (eq. (6))
$\mathbf{j}, js, jv$	indices for state mesh points (eq. (9), (15), (18))
m, M	state dimension, number of mesh points for each state (eq. (1), (9))
n	control space dimension (eq. (1))
$\mathbf{P}$	Poisson noise vector (eq. (1))
q	Poisson noise dimension (eq. (1))
r	Gaussian noise dimension (eq. (1))
s	forward time variable (eq. (1))
t, t_f, T_k	backward time variable, final time, discrete time (eq. (3), (9))
$\mathbf{u}$	control vector (eq. (1))
$\mathbf{U}_R, \mathbf{U}*$	regular control, optimal control (eq. (7), (8))
$\mathbf{U}_{min}, \mathbf{U}_{max}$	control constraint vectors (eq. (8))
$V, V*$	total cost, optimal expected total cost (eq. (3), (2))
$\mathbf{W}$	Gaussian noise vector (eq. (1))
$\mathbf{x}, X_{j_i}$	initial state vector, discrete state (eq. (2), (9))

$\mathbf{y}$	forward state variable (eq. (1))
λ_l	component of Poisson jump rate vector (eq. (6))

THE BASIC COMPUTATIONAL PROCEDURE

The integration of the PDE in (6) is backward in time, because V^* is specified finally at the final time $t = t_f$, rather than at the initial time. A summary of the discretization in state and backward time is given below:

$$\begin{aligned} \mathbf{x} \quad &\longrightarrow \mathbf{X_j} = [X_{ij_i}]_{m\times 1} = [X_{i1} + (j_i - 1)\cdot DX_i]_{m\times 1} , \\ \mathbf{j} \quad &= [j_i]_{m\times 1} , \text{ where } j_i = 1 \text{ to } M_i , \text{ for } i = 1 \text{ to } m ; \\ s \quad &\longrightarrow T_k = t_f - (k-1)\cdot DT , \text{ for } k = 1 \text{ to } K ; \\ V^*(\mathbf{X_j}, T_k) &\longrightarrow V_{\mathbf{j},k} ; \quad \mathcal{H}[V^*](\mathbf{X_j}, T_{k+\frac{1}{2}}) \longrightarrow \mathcal{H}_{\mathbf{j},k+\frac{1}{2}} ; \end{aligned} \tag{9}$$

where DX_i is the mesh size for state i and DT is the step size in backward time.

The numerical algorithm is a modification of the predictor corrector, Crank Nicolson methods for nonlinear parabolic PDEs in [5]. Modifications are made for control feedback, switch term optimization and delay term calculations. Derivatives and Poisson induced differences are approximated with an accuracy that is second order in the local truncation error $\mathcal{O}^2(DX_i)$, at all interior and boundary points. Even though the Bellman equation (6) is a single PDE, the process of solving it not only produces the optimal expected cost V^*, but also the optimal expected feedback control law $\mathbf{U}^*$. This is because the Bellman equation is a functional PDE, in which the computed regular control feeds back into the optimal cost and the optimal cost feeds back into regular control through its gradient. The nonstandard part of the algorithm is to decompose this tightly coupled analytical feedback system so that both the cost and the control can be calculated by successive iterations, such that each successive approximation of one quantity improves the next approximation of the other quantity. While our procedure may look superficially like a standard application of finite differences, it is not due to the nonstandard control features mentioned above. For these reasons, we are not aware of any other successful stochastic dynamic programming code that treats anywhere near the generality of applications that we treat and with the advanced computing techniques that we use, especially with regard to Poisson noise. Variations of this algorithm have been successfully utilized in [16] and [8]. Quadrat and his co-workers [1] discuss several algorithms for stochastic dynamic programming problems that admit stationary solutions, and describe an expert system for their solution.

Prior to calculating the values, $V_{\mathbf{j},k+1}$, at the new $(k+1)^{st}$ time step for $k = 1$ to $K-1$, the old values, $V_{\mathbf{j},k}$ and $V_{\mathbf{j},k-1}$, are assumed to be known, with $V_{\mathbf{j}0} \equiv V_{\mathbf{j}1}$. The algorithm begins with an convergence accelerating *extrapolator (x) start*:

$$V^{(\mathbf{x})}_{\mathbf{j},k+\frac{1}{2}} = \tfrac{1}{2}(3\cdot V_{\mathbf{j},k} - V_{\mathbf{j},k-1}) . \tag{10}$$

The extrapolated values are use to calculate updated values of the gradient DV, the second order derivatives DDV, the Poisson functional terms (V^* evaluated at $(\mathbf{x}+\mathbf{H})$), the regular control UR, the optimal feedback control U^*, and the spatial functional $\mathcal{H}_{\mathbf{j},k+0.5}$ of the Bellman equation. These evaluations are used in the *extrapolated predictor (xp) step*:

$$V^{(\mathbf{xp})}_{\mathbf{j},k+1} = V_{\mathbf{j},k} + DT\cdot\tfrac{1}{2}\mathcal{H}^{(\mathbf{x})}_{\mathbf{j},k+\frac{1}{2}} . \tag{11}$$

which are then used in the *predictor evaluation (xpe) step*:

$$V^{(\mathrm{xpe})}_{\mathbf{j},k+\frac{1}{2}} = \tfrac{1}{2}(V^{(\mathrm{xp})}_{\mathbf{j},k+1} + V_{\mathbf{j},k}) , \tag{12}$$

and continuing with other terms of the spatial functional $\mathcal{H}$. The evaluated predictions are used in the *corrector (xpec) step*:

$$V^{(\mathrm{xpec},\gamma+1)}_{\mathbf{j},k+1} = V_{\mathbf{j},k} + DT \cdot \mathcal{H}^{(\mathrm{xpe},\gamma)}_{\mathbf{j},k+\frac{1}{2}} \tag{13}$$

for $\gamma = 0$ to γ_{max} until the stopping criterion is met, with *corrector evaluation (xpece) step*:

$$V^{(\mathrm{xpece},\gamma+1)}_{\mathbf{j},k+\frac{1}{2}} = \tfrac{1}{2}(V^{(\mathrm{xpec},\gamma+1)}_{\mathbf{j},k+1} + V_{\mathbf{j},k}) . \tag{14}$$

The stopping criterion for the corrections is formally derived from a comparison to a predictor corrector convergence criterion for a linearized, constant coefficient PDE. A robust mesh selection method is used to determine the stopping criterion, so that only a couple of corrections are needed, except at the first time step. The proper selection of the time to state mesh ratio guarantees that the corrections for the comparison equation converge, whether the Bellman equation is parabolic-like when the Gaussian noise is present or hyperbolic-like when there is no Gaussian noise.

Current efforts are concentrated on implementing the code on the Alliant FX/8, Cray X-MP/48, Cray 2S/4-128, and the Connection CM-2 for more general multi-state and multi-control applications. In order to implement the code for arbitrary state space dimension, a more flexible data structure is needed for the problem arrays, $\mathbf{F}$, G and H, as well as for the solution arrays, V along with its derivatives and $\mathbf{U}$.

The advantages of the algorithm is that it 1) permits the treatment of general continuous time Markov noise or deterministic problems without noise in the same code, 2) maintains feedback control, 3) permits the cheap control limit to linear singular control to be found from the same quadratic cost code, and 4) produces very vectorizable and parallelizable code whose performance is described in the next section.

ADVANCED SUPERCOMPUTER OPTIMIZATION

The code for the algorithm has been developed and tested on three advanced architecture machines, the ACRF Alliant FX/8 vector multiprocessor at Argonne National Laboratory; the NCSA Cray X-MP/48 and the NCSA Cray 2S/4-128 at the University of Illinois in Urbana; the massively parallel Connection Machine CM-2 at both the ACRF and NCSA. The Alliant FX/8, with its superb concurrent outer, vector inner (COVI) parallelizing compiler, is mainly used to test for the parallelization of the code. The Cray X-MP/48, noted for its very fast pipelined processing unit, is used for the testing of small and moderate size code (less than 1 MW, where MW denotes a megaword or one million words). As the number of states grows, the problem size grows exponentially, we have to make use of the huge internal memory (up to 128 MW) of the Cray 2S/4-128 or large numbers of parallel processors on the Connection Machine CM-2.

The present code under testing has been obtained from the three-state, three-control modification of Hanson's two-state, two-control resource model [8]. Modifications have been made to the present code so that it can apply to arbitrary number of state variables and mesh points by just changing a few parameters. numbers of state variables m and mesh points M.

Initial parallelization and vectorization of the algorithm were done by what might be called the

"Machine Computational Model Method," i.e., tuning the code to optimizable constructs that are automatically recognized by the compiler, with the Alliant FX/8 vector multiprocessor [2] in mind. All inner double loops were reordered to fit the Alliant *concurrent-outer, vector-inner* (*COVI*) model. All non-short single loops were made *vector-concurrent*. Short loops became *scalar-concurrent* only. Multiple nested loops were reordered with the two largest loops innermost. A total of 37 out of 39 loops was optimized for the two-state code, two-control model with Poisson noise. Detailed results are reported in [8], [9] and [10].

The relative performance of column oriented versus row oriented code is discussed in [11]. Dongarra, Gustavson, and Karp [4] have demonstrated that loop reordering gives vector or supervector performance for standard linear algebra loops on a Cray 1 type column oriented FORTRAN environment with vector registers. However, for the stochastic dynamic programming application, the dominant loops are non-standard linear algebra loops, so that the preference for column oriented loops is not a rule, as demonstrated on the Alliant vector multiprocessor [11]. The present code under testing has up to four states and controls, with Gaussian as well as Poisson noise. This code is a general modification of the two-state, two-control model.

Vector Data Structure

In the original code, the data structure for the problem arrays, $\mathbf{F}$ and G, the solution arrays V, the derivative arrays, and the control arrays $\mathbf{U}$, depend on all the numerical node indices, $js(is)$, for all state variables. The resulting data structure takes the form:

$$F(is, js(1), js(2), \cdots, js(m)) \tag{15}$$

for each state equation, $is = 1$ to m, with the nonlinearity function used as an example. If it is assumed that there are a common number $M = M_1 = M_2 = \cdots = M_m$ of nodes per state, then $js(is) = 1$ to M points for $is = 1$ to m states. As a consequence, the typically dominant loops for the computation of the nonlinearity function F, the solution gradient DV, and similar arrays, are nested to a depth of at least $m + 1$. A typical loop will take the form:

```
      do 1 i = 1, m
         do 1 j1 = 1, M
                 :
            do 1 jm = 1, M
                    :
1              F(i,j1,j2,···,jm) = ·····
```

This state size dependent loop nest depth level of $m + 1$ inhibits the development of general multibody algorithms, especially when the state size m is incremented and the number of loops in each nest has to be changed. Also, vectorization is inhibited for compilers that vectorize only the most inner loop. As the number of states grows, the computational load will grow asymptotically like some multiple of

$$m \cdot M^m = m \cdot e^{m \ln(M)}, \tag{16}$$

i.e., the load grows exponentially in the number of states m. The exponential growth in (16) is merely a quantitative expression of *Bellman's curse of dimensionality.*

One way around this inhibiting structure is to use a *vector data structure* [12]:

$$\mathsf{FV(is,jv)} \tag{17}$$

to replace the original *hypercube type of data structure* in (15), using the nonlinearity vector as an example, such that all the numerical nodes are collected into a single vector indexed by the global state index jv, where $jv = 1$ to M^m over all state nodes.

$$jv = \sum_{i=1}^{m} (js(i) - 1) \cdot M^{i-1} + 1, \tag{18}$$

in the the case that the state mesh size, M_i has a common value of M for all i.

Both the direct mapping from the original data structure to the vector data structure and the inverse mapping are needed to compute the amplitude functions, $\mathbf{F}$, G and H, as well as the derivatives of V^*, because these quantities depend on the original formulation. The pseudo-inverse of the vector index in (18) can be shown to permit the recovery of the individual state indices by way of integer arithmetic:

$$js(is; jv) = 1 + [jv - 1 - \sum_{i=is+1}^{m} (js(i; jv) - 1) \cdot M^{i-1}]/M^{is-1}, \tag{19}$$

recursively, for $is = m$ to 1, by back substitution, with $\sum_{i=m+1}^{m} a_i \equiv 0$, as long as each state has the same number of discrete nodes M. The vector data structure of (17) to (19) results in major do loop nests of the order of 1 to 2, rather than order of $m + 1$.

A typical vector data structure loop has the form

```
      do  2  i = 1, m     !  parallel loop.
         do  2  jv = 1, M**m     !  vector loop.
                  :
2                FV(i,jv) = ......
```

resulting in collapsing the loop nest depth from $m + 1$ to a depth 2, independent of the number of states m. This is analogous to the automatic compiler technique of *loop collapsing* on the Alliant for simple loops.

Table I shows the performance of the code for $m = 3$ states and $M = 16$ nodes per state on the Alliant FX/8 at Argonne National Laboratory's ACRF. implemented and run on vector multiprocessors which will be discussed in the following two subsections.

Parallelization in Alliant FX/8

When loop 2 above is executed on multiprocessors such as the Alliant FX/8, due to the *COVI* (concurrent-outer, vector-inner) compiler optimization scheme, the i-loop will run in parallel while the jv-loop is vectorized. For machines with such architecture, the gain in speed is achieved through the full exploitation of all its processors. If the number states m is less than the maximum number of processors (the maximum number of processors is eight on the Alliant), performance ceases to improve beyond m processors as demonstrated in Table I when the number of processors p is greater than three. The speedup $S_{p,2} = T_{1,2}/T_{p,2}$ for loop 2 also levels off at roughly 2.9 in this table starting at $p = 3$. This means degradation in efficiency because a large proportion of processors available are sitting idle.

One simple modification of the loop structure will solve the problem by parallelizing and vectorizing the entire loop nest, further enhancing the performance. This is illustrated by the restructuring of loop 2 by a compiler directive in loop 3 below.

```
      do  3  jv = 1, M**m     !  vector-concurrent loop.
CVD$L NOVECTOR
```

```
              do  3  i = 1, m    !  scalar loop.
3                 FV(i,jv) = ·····
```

Due to the flexibility of the optimization scheme of the FX Fortran compiler, we can choose whichever loop we want to parallelize and vectorize by inserting suitable compiler directives, such as *CVD$L NOVECTOR* in loop 3. The i-loop is moved innermost and is forced to run in *Scalar Mode* by inserting a *CVD$L NOVECTOR* directive. The modification has two effects:

i. the outer jv-loop is forced to run in *Vector-Concurrent Mode*, hence, full parallelization of the entire work load can be achieved through self-scheduling by the compiler;

ii. moving the i-loop inner-most increases the chunk or grain size of each iteration, while overhead for parallelization and vectorization is lessened.

The modification leads to an improvement of 46% of computing time for the code running by 8 processors in the Alliant.

Table II shows the performance of the modified code, for $m = 3$ states and $M = 16$ points per state, on the Alliant FX/8 at Argonne National Laboratory's ACRF. The speedup, also given in the table, reaches a good value over six times executing on all eight Alliant processors using the form of loop 3. The last column compares the results of using loops 2 and 3 for the main stochastic dynamic programming loops and shows that the loop 3 form outperforms the loop 2 form by 1.85 times on all eight processors. Thus, the restructured loop 3 gives better load balancing that the pure vector data structure of loop 2.

Parallelization on the Cray 2

Parallelization in the Cray 2S/4-128 is done through multitasking. Basically, the compiler follows the *COVI* optimization scheme that the outer loop will run in parallel and inner loop is vectorized. In a multi-user environment such as that in NCSA, improvement through multitasking is hard to measure unless the code is run in a dedicated machine. Therefore, performance utilities such as Job Accounting (ja), are used to get an approximate measure of the CPU time and speed-up obtained.

Table III shows the performance of multitasking on the NCSA Cray 2S/4-128. Note that the timings grow drastically as either the state dimension m and the common mesh size M increase.

Performance on the Connection Machine

As the number of states increase, the performance obtained from Cray shows an exponential growth as in Table IV. Thus for a larger size problem, another solution would be to implement the problem on a massively parallel computer system. The Connection Machine CM-2 at the NCSA has $32K$ or 32,768 bit processors and one floating point processor for every 32 bit processors.

The preliminary results obtained from the Connection Machine are shown in Table IV. We implemented the problem in the Fortran 8X language with array notation extensions and two dimension data structure. Also the CM-2 directives CMF$LAYOUT and CMF$ALIGN overlay different size of arrays, in order to reduce the internal communication time and hence improve the performance. The program is compiled with $-O$ option and run with at most $32K$ data processors in single precision. The preliminary results on the CM-2 indicate that when the problem size increases, the Connection Machine computes the problem with relatively small increase in the execution time. For instance, for the case nx=4 states, when mesh size per state increases from M=8 to M=16, the execution time increases by about 2.1 times for *Real Time* (sum of CM-2 time and the time on the front-end computer) and 5.8 times for *CM*

Time in Table IV, while on the Cray the time increases by about 23.8 times, according to Table III, which is much larger than for the CM-2.

It must be noted that the Cray and the CM-2 have different computational structures and our current Fortran 8X program is translated from our Cray algorithm and data structure. A further goal will be to modify the algorithm and data structure so that the performance on the CM-2 will be competitive with the performance on the Crays in an absolute sense.

Computation for Boundary Points

The computation of the solution gradient DV and the array of second derivatives DDV, which is carried out in the subroutine $GETDV$, requires different algorithms for the interior nodes and the boundary nodes. Due to the complexity and generality of the underlying stochastic dynamical system, the boundary values cannot be specified in general, but must be calculated from the Bellman Equation (6) itself, except for the most trivial boundaries and processes. Use of the Bellman Equation at the boundaries, makes the algorithm segments for updating the boundary values quite different from the interior values in order to maintain the same order of error as at the interior points, i.e., to avoid *numerical pollution* of the order, $\mathcal{O}(DX)^2$, at the interior points. When the vector data structure is used, the boundary nodes ($js(is) = 1$ and M) are scattered throughout the data arrays $FV(is, jv)$. Due to this nonuniform distribution of the boundary nodes, a time-consuming nested if-then-else loop has to be used in the original $GETDV$, which greatly degrades the computation speed. For the current testing code with $m = 3$ and $M = 16$, $GETDV$ takes 34% of the running time in the Alliant runs and 30% in the Cray X-MP runs. One way to alleviate this degradation is by *homogeneous global computation* and then separate *recorrection* for the boundary points.

Since the proportion of boundary nodes is generally small compare with internal nodes ($2/M$ for M mesh points per state) and all of them can be extracted explicitly from the inverse vector index (19). Hence, we can pass the whole data array through a homogeneous computation first, taking all points to be internal nodes, then recorrect the boundary nodes outside the main loop. Artificial or redundant points are added to prevent overwriting valid data, and it will be seen the resulting small addition to the memory by the use of artificial points is worth the benefit in improved performance.

Table V compares the performance for the old and new forms of $GETDV$ for different mesh points M run on the CRAY X-MP. A faster run time for the new version of 1.45 times the old version and a saving of up to 31% of running time is exhibited.

MEMORY REQUIREMENTS

Since the memory requirements grows exponentially with increases in the state variable from Bellman's *curse of dimensionality* (16), a machine with large internal memory is needed for the large state variable case. For the sake of a uniform comparison, all the testings were carried out on the NCSA Cray 2S/4-128, which possesses a huge internal memory (up to 128 MW). Table VI summarizes the memory requirements for different test codes. Also in Table VI, the memory in words is compared to the order of magnitude of the Bellman's *curse of dimensionality* term in (16). The approximate asymptote for large state dimensions m or very fine state meshes M is about 12, which gives the effective number of major loops of nest depth $m + 1$. The CPU time measurements in Table III have similar exponential growth characteristics to that of memory requirements.

CONCLUSIONS

Stochastic dynamic programming can be optimized for the a moderate and perhaps larger number of state variables using a vector multiprocessor. Loop collapsing using a vector data structure, compiler

directives making possible more efficient loop reordering, and homogeneous global computation making boundary value computation more efficient, all help obtain superior optimization of the stochastic dynamic programming code. Parallelization, vectorization, large memories, and other supercomputing features are important in solving larger state space problems In order to handle a large number of state variables, a large number of parallel processors with extremely large memory would be desirable, but Bellman's *curse of dimensionality* appears to very much weakened. Computation with massively parallel processors, like the Connection Machine CM-2, is still preliminary, but shows promise for larger state spaces. These techniques are generally applicable to other vector and parallel computers. Our general code is essentially valid for general Markov noise in continuous time, feedback control, nonlinear control and the cheap control limit.

REFERENCES

1. M. Akian, J. P. Chancelier and J. P. Quadrat, *Dynamic programming complexity and application*, in **Proc. 27th IEEE Conf. on Decision and Control, Vol. 2**, pp. 1551-1558, Dec. 1988.

2. Alliant, **FX/FORTRAN Programmer's Handbook**, Alliant Computer Systems Corporation, Acton, Mass., 1985.

3. R. E. Bellman, **Adaptive Control Processes: A Guided Tour**. Princeton: Princeton University Press, 1961.

4. J. J. Dongarra, F. G. Gustavson, and A. Karp, *Implementation of linear algebra algorithms of dense matrices on a vector pipeline machine*, **SIAM Rev.**, vol. 26, pp. 91-112, 1984.

5. J. Douglas, Jr., and T. DuPont, *Galerkin methods for parabolic equations*, **SIAM J. Num. Anal.**, vol. 7, pp. 575-626, 1970.

6. **Future Directions in Control Theory: A Mathematical Perspective**, W. H. Fleming, Chairman. Philadelphia: Society for Industrial and Applied Mathematics, 1988.

7. I. I. Gihman and A. V. Skorohod, **Controlled Stochastic Processes**. New York: Springer-Verlag, 1979.

8. F. B. Hanson, *Bioeconomic model of the Lake Michigan alewife fishery*, **Can. J. Fish. Aquat. Sci.**, vol. 44, Suppl. II, pp. 298-305, 1987.

9. F. B. Hanson, *Computational dynamic programming for stochastic optimal control on a vector multiprocessor*, **Argonne National Laboratory, Mathematics and Computer Science Division Technical Memorandum ANL/MCS-TM-113**, June 1988, 26 pages.

10. F. B. Hanson, *Computational dynamic programming on a vector multiprocessor*, **IEEE Trans. Automat. Contr.**, 12 pages, to appear, 1990.

11. F. B. Hanson, *Parallel computation for stochastic dynamic programming: Row versus column code orientation*, in **Proceedings 1988 Conference on Parallel Processing, Vol. III Algorithms and Applications**, D. H. Bailey, Editor. University Park: Pennsylvania State University Press, 1988, pp. 117-119.

12. F. B. Hanson, *Stochastic dynamic programming: Advanced computing constructs*, **Proc. 28th IEEE Conf. on Decision and Control, Vol. I**, Dec. 1989, pp. 901-903.

13. F. Hanson and D. Ryan, *Optimal harvesting with density dependent random effects*, **Natural Resource Modeling**, vol. 2, No. 3, pp. 439-455, 1988.

14. D. Ludwig, *Optimal harvesting of a randomly fluctuating resource I: Application of perturbation methods*, **SIAM J. Appl. Math.**, vol. 37, pp. 166-184, 1979.

15. C. D. Polychronopoulos, **Parallel Programming and Compilers**. Boston: Kluwer Academic Publishers, 1988.

16. D. Ryan and F. B. Hanson, *Optimal harvesting of a logistic population in an environment with stochastic jumps*, **J. Math. Biol.**, vol. 24, pp. 259-277, 1986.

Table I: Timings on the Alliant with vector data structure for loop 2 with 3 states and 16 nodes per state.

Number of Processors p	User CPU Time (seconds) $T_{p,2}$	Speedup $S_{p,2}$ $T_{1,2}/T_{p,2}$
1	83.68	1.00
2	56.23	1.49
3	30.07	2.78
4	29.24	2.86
5	29.20	2.87
6	29.24	2.86
7	28.77	2.91
8	28.64	2.92

Table II: Timings on the Alliant with order and directive modified loops for loop 3 with 3 states and 16 nodes per state.

Number of Processors p	User CPU Time (seconds) $T_{p,3}$	Speedup $S_{p,3}$ $T_{1,3}/T_{p,3}$	Improvement Ratio $T_{p,2}/T_{p,3}$
1	95.32	1.00	0.88
2	49.00	1.95	1.15
3	34.61	2.75	0.87
4	25.74	3.70	1.14
5	22.66	4.21	1.29
6	18.82	5.06	1.55
7	17.78	5.36	1.62
8	15.44	6.17	1.85

Table III: CPU time (seconds) for different state dimensions and different mesh sizes on the Cray 2S/4-128 with Multitasking.

State Variables m	Mesh Points M					
	8		16		32	
	T	Sp	T	Sp	T	Sp
2	0.033	1.60	0.130	3.66	0.685	3.54
3	0.104	3.53	1.169	2.86	24.626	1.54
4	2.527	3.14	60.151	1.73	2338.290	2.10

Table IV: CPU time (seconds) for different state dimensions and different mesh sizes on the Connection Machine CM-2 (CM Time) and front-end (Real Time).

State Variables m	Mesh Points M					
	8		16		32	
	Real Time	CM Time	Real Time	CM Time	Real Time	CM Time
3	10.97	2.79	21.51	5.83	52.53	30.14
4	36.02	11.42	76.01	66.41	—	—

Table V: Performance comparison of the old and new forms of GETDV on the Cray X-MP/48 for loop 3 with 3 states.

Number of Mesh Points M	User CPU Time (seconds)		Improvement Ratio T_{old}/T_{new}	Per Cent Savings $100(1 - T_{new}/T_{old})$
	Old GETDV T_{old}	New GETDV T_{new}		
8	0.142	0.098	1.45	31.
16	2.093	1.523	1.37	27.
24	10.378	7.679	1.35	26.
32	32.725	24.234	1.35	26.
40	78.510	58.467	1.34	26.

Table VI: Memory requirements for different state dimensions and different mesh sizes on the Cray 2S/4-128.

State Variables m	Mesh Points M			Mesh Points M		
	8	16	32	8	16	32
	Memory (MW)			Words/$m \cdot M^m$		
2	0.13	0.14	0.16	1000.	270.	78.
3	0.15	0.28	1.30	98.	23.	13.
4	0.34	3.28	50.23	21.	13.	12.

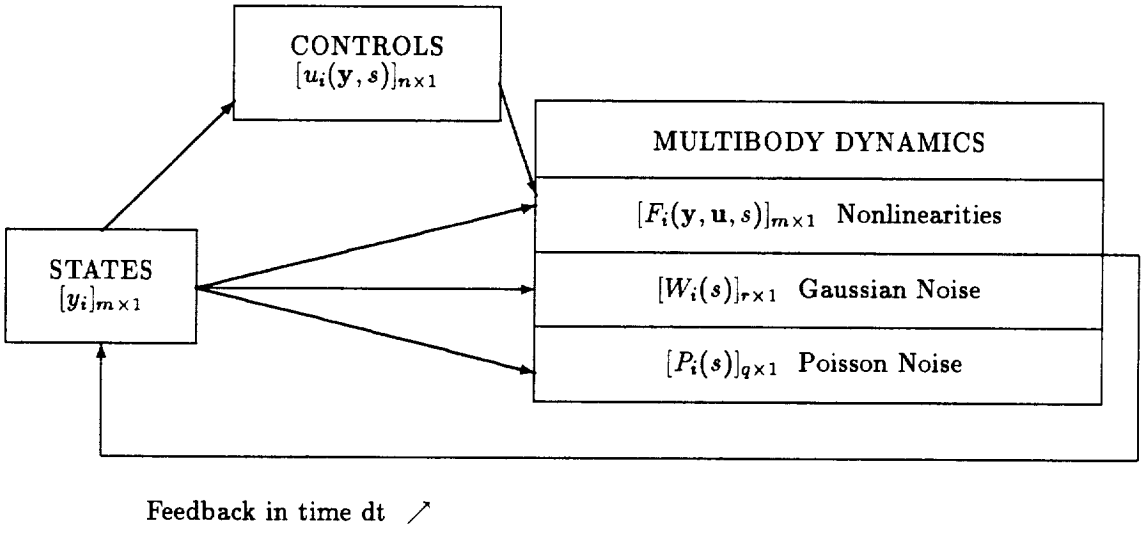


Figure 1: The stochastic multibody system with feedback.

A Fast, Reliable Algorithm for Computing Frequency Responses of State Space Models

Matt Wette
Jet Propulsion Laboratory, Caltech

Abstract

Computation of frequency responses for large order systems described by time-invariant state space systems often provides a bottleneck in control system analysis. In this talk we show that banding the A-matrix in the state space model can effectivly reduce the computation time for such systems while maintaining reliability in the results produced.

Introduction to the Problem

Consider the following realization of some transfer function $G(j\omega)$:

$$\dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t) + Du(t)$$

where $x \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, and $y(t) \in \mathbb{R}^p$. The relationship of the realization to the transfer function is given by

$$G(j\omega) = C(j\omega I - A)^{-1}B + D$$

In control system design the computation of the frequency response plays an important role in frequency-based design methods. For medium sized problems the order of $x(t)$ may be in the hundreds while $G(j\omega_k)$ must be computed for hundreds of values of ω_k. It has not been uncommon for a frequency response calculation to require hours of CPU time. Thus, efficient and reliable algorithms for this computation are needed for handling large order systems.

Typical Approach

A typical approach to computing frequency responses for state space systems is to first perform a state transformation on the realization to bring the A-matrix into some reduced form and then solve the appropriate system of linear equations for each frequency point.

Computational Issues

The above algorithm for computing frequency responses involves two issues: efficiency and sensitivity. A potential bottleneck in computing frequency responses is the solution of $(j\omega_k I - \tilde{A})X = B$ for X. Efficient computation is accomplished by reducing A to some form $\tilde{A}$ which allows efficient solution of the above equation. Another issue is that of sensitivity. The transformation process takes place in finite precision arithmetic and hence will change, to some degree, the properties of the transfer function which the realization represents. It is important, therefore, to consider the numerical properties of the transformation.

Sensitivity of the Transformation

The effect of numerical computations in the presense of finite precision arithmetic can be treated in terms of sensitivity of the coefficient matrices. Tranformations which do not increase sensitivity to state transformations are termed *well conditioned*. Ill-conditioned transformations can and usually do significantly increase the sensitivity of the coefficient matrices to small perturbations. Presense of this sensitivity is often an indication of a numerically unstable algorithm.

Efficient Solutions to $(j\omega_k I - A)X = B$

As stated, efficient solution of the above equation is usually accomplished by reducing A to some special form. Consider the case where $m = 1$ (i.e., $B \in \mathbb{R}^{n\times 1}$). Then for A in general form, solution of the equation requires $\mathcal{O}(n^3)$ floating point operations (or flops). For $\tilde{A}$ in Hessenberg form or Schur form, where $\tilde{A}$ is "nearly" upper triangular, solution of the equation requires $\mathcal{O}(n^2)$ flops. The

transformation to produce these forms is known to be extremely well-conditioned. For $\tilde{A}$ in diagonal form or Jordan canonical form, solution of the equation requires $\mathcal{O}(n)$ flops. However, the transformation producing these forms may be very ill-conditioned, leading to a reduced form which no longer accurately represents the tranfer function of interest. A good compromise is to seek some compromise between the upper-triangular and diagonal forms. This leads reduced forms in which $\tilde{A}$ is upper-triangular and block diagonal or upper-triangular and banded. We feel the banded form is a better strategy since it is a simpler structure to work with. The banded matrix is characterized by its order and bandwidth; the block diagonal form is characterized by order, block sizes, and block order. Due to the simpler structure of the banded matrix, algorithms based on banded matrices can be adopted to vector hardware architectures in a nicer way.

A New Banding Algorithm

The new banding algorithm uses several steps. First the matrix A is reduced to real Schur, or quasi-upper-triangular, form A_s. Then an ordering algorithm is applied to order the eigenvalues appearing on the diagonal of A_2 in a way that will aid the next step in producing a small bandwidth. The transformations associated with the first two processes is very well conditioned. The third step involves examining the properties of the eigenvalues to determine a "good" bandwidth *a priori*. A "good" bandwidth is one for which the condition number of T is small. Next the matrix is reduced to banded form, A_b, using a series of operations to eliminate off diagonal elements. The operations are accumulated in a matrix T. If T is found to be ill-conditioned, then the tolerance for Step 3 is tighned and Steps 3 and 4 are repeated. Finally, the matrix A_b is brought to complex, upper-triangular, banded form using a series of Givens transformations. We note that the transformations used in Step 4 are scaled to provide reduction in their condition numbers.

An Illustration

The figure shown illustrates the banding algorithm. The first operation shows the effect of bringing the system to Schur form. After the matrix has been brought to Schur form, the matrix is analyzed to determine a "good" bandwidth. Here we choose a bandwidth of 2. The second set of operations shows how the algorithm reduces a diagonal of the matrix. The third set of operations shows how the remaining diagonals are eliminated to produce the final upper-triangular, banded matrix.

Test Case

The algorithm described has been coded into Fortran and installed into our Pro-Matlab implementation using the Pro-Matlab MEX facility. We chose as a test set a set of single input, single output systems with state order ranging from 20 to 80. Matrix coefficients were generated from a random number generator. For each case we computed 200 frequency points. The table shows times for the Pro-Matlab **bode** function versus times for our **bodeq** function. As one can see, the new algorithm reduced the computation time from 75 to 88 percent.

Extensions and Future Work

The algorithm has also been applied to time simulation of linear, time-invariant systems. The banding strategy and algorithm could be extended to generalized state space systems. In this case, we would band the A and E matrices simultaneously. Another possible area of future work would be production of better banding algorthms. The current algorithm is bases on solution of Sylvester equations and has a limitation: the algorithm cannot band systems well when all eigenvalues are very closely spaced. It should be possible to band these matrices using different algorithms.

Summary

In summary, we have developed a new algorithm for computing frequency responses of state space models. In this development, we have taken into account the two prime computational issues: efficiency and sensitivity. We showed that the algorithms worked on a test problem and was able to reduce computational time considerable without a notable cost in accuracy. Finally, we proposed that the banding strategy may provide further application in control system design.

JPL

A Fast, Reliable Algorithm for Computing Frequency Responses of State Space Models

Matt Wette
Jet Propulsion Laboratory, Caltech

JPL

- Introduction to the Problem
- Typical Approach
- Computational Issues
- A New Banding Algorithm
- Test Case
- Extensions and Future Work
- Summary

JPL

Introduction to the Problem

Consider a transfer function $G(j\omega)$ with state space realization given by

$$\dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t) + Du(t)$$

where $u(t) \in \mathbb{R}^m$, $x(t) \in \mathbb{R}^n$, and $y(t) \in \mathbb{R}^p$.

$G(j\omega)$ is associated with $\{A, B, C, D\}$ through

$$G(j\omega) = C(j\omega I - A)^{-1}B + D$$

Problem: desire $G(j\omega_k)$ for many (hundreds) of values of $j\omega_k$ where $n < 200$ (medium order systems).

JPL

Typical Approach

1 Transform the system realization:

$$\{A, B, C, D\} \xrightarrow{T} \{\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}\} := \{T^{-1}AT, T^{-1}B, CT, D\}$$

2 For each ω_k do

a) solve $(j\omega_k I - \tilde{A})X = \tilde{B}$ for X

b) compute $G(j\omega_k) = \tilde{C}X + \tilde{D}$

Compuatational Issues

- Solution of $(j\omega_k I - \tilde{A})X = \tilde{B}$ must be efficient. This is usually the limiting factor.
- Given the presence of finite precision arithmetic $\{\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}\}$ must be an accurate realization of $G(j\omega)$.

Sensitivity of the Transformation

Consider

$$\{A+\Delta A, B+\Delta B, C+\Delta C, D+\Delta D\} \xrightarrow{T} \{\tilde{A}+\Delta\tilde{A}, \tilde{B}+\Delta\tilde{B}, \tilde{C}+\Delta\tilde{C}, D+\Delta D\}$$

Then we have

$$\kappa(T)^{-2}\frac{\|\Delta A\|}{\|A\|} \leq \frac{\|\Delta\tilde{A}\|}{\|\tilde{A}\|} \leq \kappa(T)^2\frac{\|\Delta A\|}{\|A\|}$$

$$\kappa(T)^{-1}\frac{\|\Delta B\|}{\|B\|} \leq \frac{\|\Delta\tilde{B}\|}{\|\tilde{B}\|} \leq \kappa(T)\frac{\|\Delta B\|}{\|B\|}$$

$$\kappa(T)^{-1}\frac{\|\Delta C\|}{\|C\|} \leq \frac{\|\Delta\tilde{C}\|}{\|\tilde{C}\|} \leq \kappa(T)\frac{\|\Delta C\|}{\|C\|}$$

where $\kappa(T) := \|T\|\|T^{-1}\|$ and $\|\cdot\|$ is some consistent norm.

Efficient Solutions to $(j\omega_k I - \tilde{A})X = \tilde{B}$

- A a general matrix $\Rightarrow \mathcal{O}(n^3)$ flops/ω_k
- $\tilde{A}$ an upper Hessenberg or Schur matrix $\Rightarrow \mathcal{O}(n^2)$ flops/ω_k
- $\tilde{A}$ a matrix in diagonal form $\Rightarrow \mathcal{O}(n)$ flops/ω_k
- $\tilde{A}$ a matrix in Jordan canonical form $\Rightarrow \mathcal{O}(n)$ flops/ω_k
- $\tilde{A}$ a block-diagonal matrix $\Rightarrow \mathcal{O}(n^l)$ flops/ω_k, where $1 \leq l \leq 2$
- $\tilde{A}$ a *banded matrix* $\Rightarrow \mathcal{O}(n \cdot bw)$ flops/ω_k where bw is the *bandwidth*, $0 < bw < n$

A New Banding Algorithm

Given a full general A matrix, produce A_b where A_b is upper triangular and banded.

The algorithm is essentially

1 Reduce A to upper real Schur form, A_s

2 Order the eigenvalues appearing on diagonal blocks of A_s to produce A_o

3 Analyze A_o to determine a "good" bandwidth (to make $\kappa(T)$ small)

4 Uses an algorithm based on solving Sylvester equations to band A_o, producing A_b

5 Convert the quasi-upper triangular matrix A_b to complex, upper triangular form.

Step 4 uses transformations of the form

$$T_{i,j} = \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & x_{i,j} & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & d_2 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & d_4 & 0 \\ 0 & 0 & 0 & 0 & I \end{bmatrix}$$

An Illustration

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{bmatrix} \xrightarrow{Q_s} \begin{bmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{bmatrix}$$

$$\begin{bmatrix} \times & \times & \otimes & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{bmatrix} \xrightarrow{T_{1,3}} \begin{bmatrix} \times & \times & 0 & \times & \times \\ 0 & \times & \times & \otimes & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{bmatrix} \xrightarrow{T_{2,4},T_{3,5}} \begin{bmatrix} \times & \times & 0 & \times & \times \\ 0 & \times & \times & 0 & \times \\ 0 & 0 & \times & \times & \otimes \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{bmatrix}$$

$$\begin{bmatrix} \times & \times & 0 & \otimes & \times \\ 0 & \times & \times & 0 & \otimes \\ 0 & 0 & \times & \times & 0 \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{bmatrix} \xrightarrow{T_2} \begin{bmatrix} \times & \times & 0 & 0 & \otimes \\ 0 & \times & \times & 0 & 0 \\ 0 & 0 & \times & \times & 0 \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{bmatrix} \xrightarrow{T_3} \begin{bmatrix} \times & \times & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 \\ 0 & 0 & \times & \times & 0 \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{bmatrix}$$

Test Case

Single-Input, Single-Output Model with Random Coefficients
Pro-Matlab **bode** function versus banded **bodeq** function

pts.	n	bode (sec)	bodeq (sec)	bw	$\kappa(T)$
200	20	39.9	4.8	3	250
200	30	81.1	10.6	3	350
200	40	139.8	24.0	13	45
200	50	211.7	41.6	17	38
200	60	300.6	64.8	20	66
200	70	407.7	104.6	31	27
200	80	527.1	135.1	32	67

Reduction in time from 75% to 88%

Extensions and Future Work

- Application to time simulation of $\{A, B, C, D\}$
- Application to extended (or generalized) models:

$$E\dot{x} = Ax + Bu\,, \quad y = Cx + Du$$

- Better banding algorithms: Banding strategy has good potential but current algorithm has some limitations.

Summary

- Developed new algorithm for computing frequency responses of state space systems.
- The algorithm provides a method for trading off the two computational issues at hand: sensitivity and efficiency.
- The algorithm was shown to provide large saving in computational time on a set of test problems.
- The strategy has some potential for other applications on medium order models.

COUPLED RICCATI EQUATIONS FOR COMPLEX PLANE CONSTRAINT

KRISTIN M. STRONG AND JOHN R. SESAK
Lockheed Missiles and Space Co., Inc.
Sunnyvale, Calif. 94089-3504

ABSTRACT

A new LQG design method is presented which provides prescribed imaginary-axis pole placement for optimal control and estimation systems. This procedure contributes another degree of design freedom to flexible spacecraft control: Current design methods which interject modal damping into the system tend to have little affect on modal frequencies, i.e. they predictably shift open-loop plant poles horizontally in the complex plane to form the closed-loop controller or estimator pole constellation, but make little provision for vertical (imaginary-axis) pole shifts. Imaginary-axis shifts which reduce the closed-loop modal frequencies (the bandwidth) are desirable since they reduce the sensitivity of the system to noise disturbances. The new method drives the closed-loop modal frequencies to predictable (specified) levels--frequencies as low as zero rad/sec (real-axis pole placement) can be achieved. The design procedure works through rotational and translational destabilizations of the plant, and a coupling of two *independently*-solved algebraic Riccati equations through a structured state-weighting matrix. Two new concepts, *gain transference* and *Q-equivalency*, are introduced and employed in the design process.

1. INTRODUCTION

Multi-input, multi-output systems, such as those encountered in flexible spacecraft control, are often approached with modern optimal control techniques which conveniently generate closed-loop system gain matrices for simultaneous multi-loop closures. However, modern optimal control, as presented in most textbooks, is not a complete control system design methodology. The major problems of translating control system performance requirements, bandwidth constraints, and compensator robustness constraints into the performance index have not been fully developed [1]. The result is a control system design methodology that is iterative and empirical. An approach to solving these problems and de-empiricizing the design process is to use structured performance index (SPI) constraints [2]. SPI constraints may be defined as structured performance index weighting matrices which constrain the weighted variables to approach desired predefined directions and values in the state space as the weighting matrix entries approach infinity. This is in contrast to generalized constraints for which the weighted variables approach zero as the weighting matrix entries approach infinity. To employ structured constraints, and avoid the application of generalized constraints, the weighting matrices for the SPI must be less than full rank. The potential usefulness of the SPI approach is apparent: An appropriately-structured performance index can

drive state variables in predictable directions thereby achieving a desired performance and bandwidth objective. SPI's can provide a non-empirical means of constraining the controller, estimator, and compensator dynamics--the latter is critical for closed-loop system robustness.

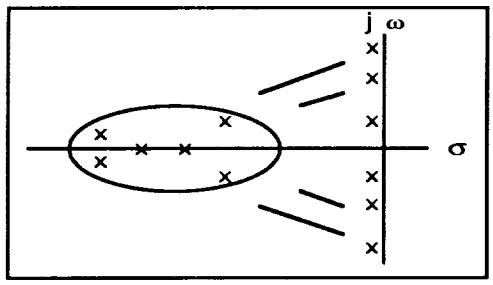


Figure 1. Design Objective: Complex Plane Constraint

The overall design objective for the controller/estimator utilizing the SPI approach is illustrated in Fig. 1: The gain matrices generated through the SPI translate the open-loop poles to some prescribed closed-loop boundary in the complex plane. Similarly the compensation dynamics are constrained to reduce closed-loop system sensitivity. Note that to achieve this objective, two degree-of-freedom control is required over each controlled mode, i.e. poles require movement in two dimensions in the complex plane, both horizontally (along the real-axis) and vertically (along the imaginary-axis).

In the next section we review SPI design methods for prescribed real-axis constraint in the optimal control, estimation, and compensation systems, and introduce a coupled Riccati equation design technique for prescribed imaginary-axis constraint.

2. SPI DESIGN METHODS

Currently a well-known performance index exists for prescribed *real-axis* pole translations in the optimal controller and estimator systems [1]: In the "alpha-shift" technique shown in Fig. 2, the standard Linear Quadratic Gaussian (LQG) performance index is augmented with an exponential weighting. This exponential weighting guarantees that the quadratic terms in the performance index decay with at least a rate of 2α so that the performance index remains finite over the infinite interval. The result is a guaranteed stability margin--all closed-loop poles lie to the left of the -2α-line in the complex plane.

The design procedure with the alpha-shift technique is straight-forward: $[+\alpha I]$ is appended to the nominal plant dynamics, A. This tends to destabilize the plant. Optimal control theory is applied

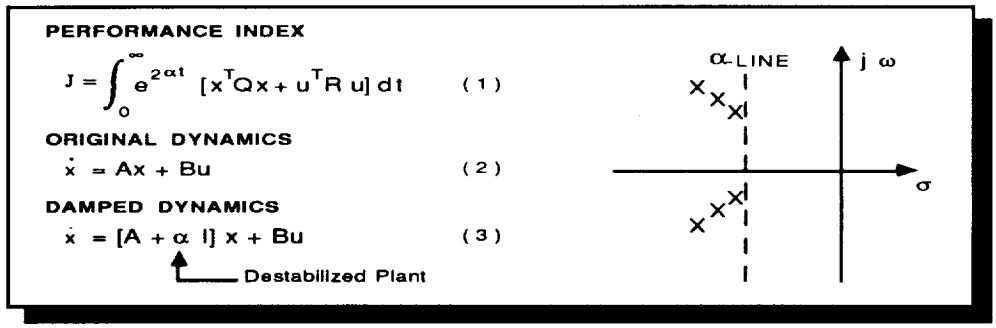


Figure 2. The Alpha-Shift Technique Provides a Prescribed Real-Axis Pole Translations [Anderson & Moore]

and state feedback gains are generated for the destabilized plant, characterized by $[A+\alpha I]$, which are guaranteed to stabilize it. When these gains are applied to the *nominal* plant, A, the closed-loop poles have real parts of -2α. This technique provides horizontal (real-axis) translation of the plant poles from their open to closed-loop positions.

For the compensator, predictable *real-axis* pole translations are also possible through *indirect* SPI design techniques which structure control and observation constraints [3]. These constraints tend to normalize the control and observation effort thereby providing indirect control over compensator poles, bandwidth, and closed-loop singular values.

Prescribed *imaginary-axis* pole translations in the optimal control and estimation systems are the focus of this paper: SPI design techniques are presented which drive the modal frequencies of the closed-loop system to desired levels. *Conceptually*, prescribed *imaginary-axis* pole placement may be considered to be composed of a 90 degree rotation, a vertical translation, and a stabilization of the open-loop plant poles as shown in Fig. 3. Stabilization is achieved by generating a stabilization matrix for the plant in rotated space and applying it to a standard alpha-shift design through a SPI. Using the stabilization matrix from one optimal design process and applying it to another couples two algebraic Riccati equations (ARE's) together.

The next section introduces two key concepts, gain transference and Q-equivalency, that are critical to the development of the SPI for Riccati equation coupling. This is followed by an outline of the actual design steps required for prescribed imaginary-axis pole placement.

3. DESIGN PROCEDURE: PRESCRIBED IMAGINARY-AXIS POLE PLACEMENT

The design procedure for prescribed imaginary-axis pole placement employs a SPI that couples two ARE's together. Gain transference and Q-equivalency are important to understanding the development of this SPI.

Gain transference involves designing optimal gains for one plant and applying them to another, indirectly-related, plant. As shown in Fig. 4, optimal regulator theory is applied to system 1, generating optimal gains $-R^{-1}B^{T}P_{1}$. A closed-loop state feedback system is formed for system 2 with these gains. (Note that system

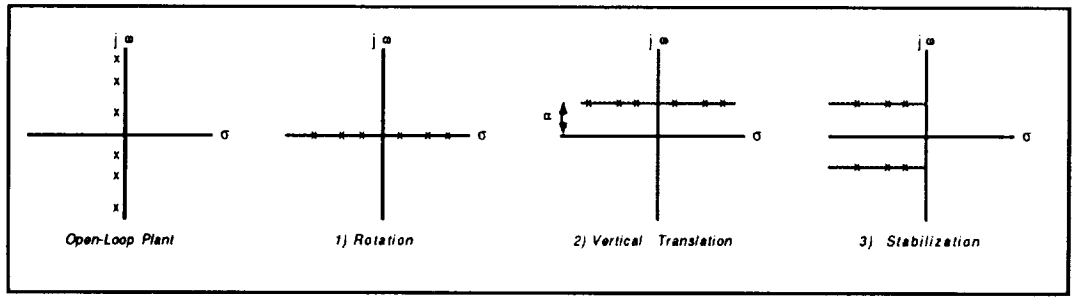


Figure 3. Conceptual Development: Prescribed Imaginary-Axis Pole Placement

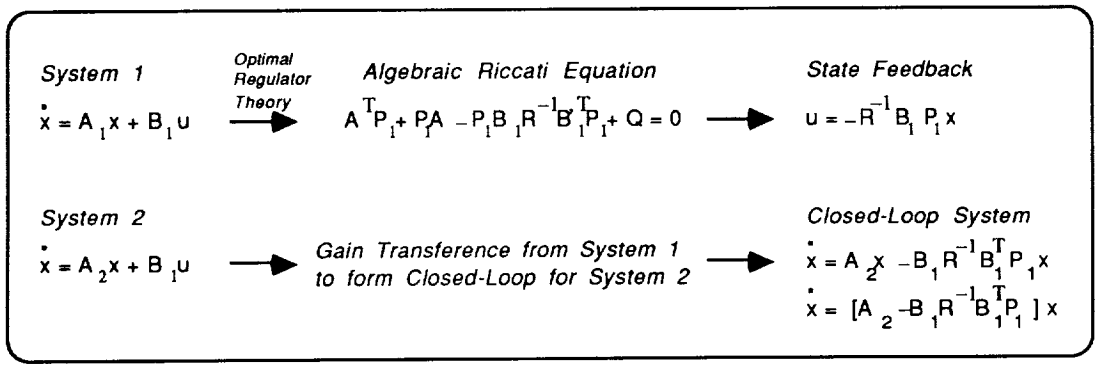


Figure 4. Gain Transference Theory

2 has identical input vectors as system 1.) The optimal gains generated for system 1 have been transferred to system 2 to form the closed-loop system.

The utility of gain transference lies in its harmonic-restructuring capability. The harmonic structure of the closed-loop system $[A_2 - B_1 R^{-1} B_1^T P_1]$ can be strongly influenced by the harmonic structure of A_1. In the design procedure for prescribed imaginary-axis pole placement, optimal gains, P_1, are generated for the plant in a rotated space, A_1. When these gains are transferred to the nominal plant, A_2, for state feedback, the closed-loop system takes on the harmonic characteristics of the plant in a rotated space. The state feedback transforms the nominal plant to rotational space--a key step in achieving prescribed imaginary-axis pole placement.

Q-equivalency, the other concept central to the design procedure, involves expanding and collecting terms in an ARE to indirectly generate a state-weighting matrix. An example of the concept is shown in Eq. 1 for the ARE employed in the alpha-shift technique.

$$(A+\alpha I)^T P + P(A+\alpha I) - PBR^{-1}B^T P + Q = 0$$

$$A^T P + PA - PBR^{-1}B^T P + 2\alpha IP = 0$$

$$Q_{eq} = 2\alpha IP$$
$$= -(A^T P + PA - PBR^{-1}B^T P) \qquad (1)$$

The alpha terms are expanded and collected to form a Q-equivalent matrix equal to $2\alpha IP$. In a SPI, Q_{eq} is a state-weighting matrix that will generate the same optimal gains for the *nominal* plant, as those generated through the ARE for the *alpha-shifted* plant. This concept is used in the design procedure to couple two ARE's together: *A Q_{eq} matrix for the ARE in rotational space is used as the state-weighting matrix for an ARE in translational (alpha-shifted) space.*

An overview of the actual design steps that employ the concepts of gain transference and Q-equivalency are illustrated in Fig. 5 and described below:

1) Rotational Plant Destabilization. A simple matrix transformation of the plant rotates poles circularly from their open-loop positions to the real-axis. This removes all harmonic components from the rotated plant dynamics. Half of the rotated

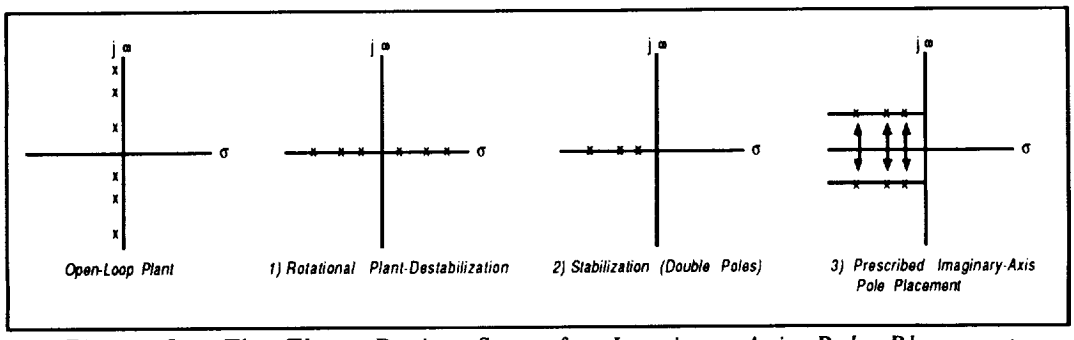


Figure 5. The Three Design Steps for Imaginary-Axis Pole Placement

plant poles in this space are unstable.

2) *Rotational Plant Stabilization.* A stabilization matrix is generated for the rotationally-destabilized plant through a SPI using standard optimal regulator design methods. Unstable right-hand-plane (RHP) poles at $\{+\omega_1, +\omega_2,...,+\omega_n\}$ are moved to the left-hand-plane (LHP) to positions of $\{-\omega_1, -\omega_2,...,-\omega_n\}$, respectively. The stabilization matrix does not affect the stable LHP poles at $\{-\omega_1, -\omega_2,...,-\omega_n\}$. The resulting closed-loop system has *double poles* at each modal frequency in the LHP.

3) *Prescribed Imaginary-Axis Pole Placement.* The stabilization matrix generated for the rotationally-destabilized plant is used in a SPI to transform an alpha-shift design to rotational space. The value of α determines the closed-loop modal frequencies, i.e. α prescribes the amount of imaginary-axis pole translation from the *real-axis*.

We now present details of the prescribed imaginary-axis pole placement design procedure for optimal control and estimation systems. The three steps outlined above are expanded and applied to a low-order system to illustrate their effects. The complete design procedure is then developed and its application to flexible spacecraft control is illustrated in a numerical example.

DESIGN STEP 1: Rotational Plant-Destabilization

To introduce rotational plant-destabilization we compare it graphically to the alpha-shift technique. As shown in Fig. 6, alpha-shifted plant-destabilization is accomplished via a horizontal translation of the poles into the right-half of the complex plane. Rotational plant destabilization occurs with circular rotations of the open-loop poles to the real-axis.

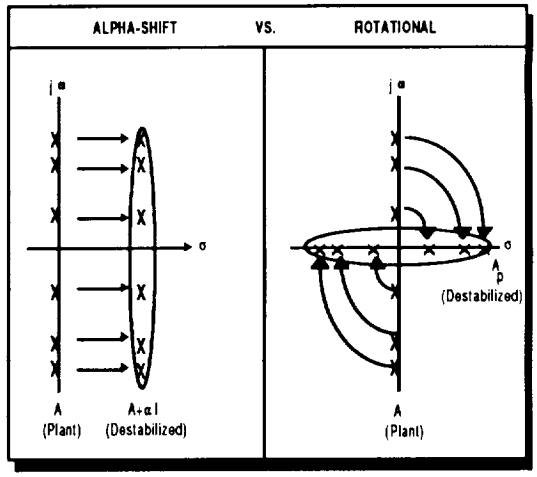


Figure 6. Plant Destabilization Techniques

The transformation matrix that accomplishes this destabilization of the plant is described below for an n-mode system. If the plant matrix, A, in block-diagonal form is:

$$A = \begin{bmatrix} \begin{matrix} 0 & 1 \\ -\omega_1^2 & 0 \end{matrix} & & & \\ & \begin{matrix} 0 & 1 \\ -\omega_2^2 & 0 \end{matrix} & & \\ & & \ddots & \\ & & & \begin{matrix} 0 & 1 \\ -\omega_n^2 & 0 \end{matrix} \end{bmatrix}_{2n \times 2n}$$

then the rotationally-destabilized plant matrix, A_r, is defined to be:

$$A_r \triangleq AT$$

where

$$T = -\begin{bmatrix} \begin{matrix} 0 & 1/\omega_1 \\ \omega_1 & 0 \end{matrix} & & & \\ & \begin{matrix} 0 & 1/\omega_2 \\ \omega_2 & 0 \end{matrix} & & \\ & & \ddots & \\ & & & \begin{matrix} 0 & 1/\omega_n \\ \omega_n & 0 \end{matrix} \end{bmatrix}$$

and $A_r = \begin{bmatrix} -\omega_1 & & & \\ & \omega_1 & & \\ & & -\omega_2 & \\ & & & \omega_2 \end{bmatrix}$ (continued)

with eigenvalues $\lambda(\pm\omega_1, \pm\omega_2, \ldots, \pm\omega_n)$. We note that this set of eigenvalues is not unique to A_r. Other transformations of the plant will produce *block-diagonal* or *off-diagonal* matrices with equivalent eigenvalue sets. These matrices may be used in lieu of A_r in designing the controllers and estimators, but the closed-loop system bandwidths tend to be larger than those designed with A_r. A_r's diagonal structure provides the smaller bandwidth controller for this design procedure, and we use this structure in the following two-mode example illustrating rotational destabilization, and in the flexible spacecraft example at end of the paper.

Given System: $\omega_1 = 10$ rad/sec
$\omega_2 = 15$ rad/sec

One actuator and collocated sensor corresponding to torque actuation and velocity sensing are employed.

$$A = \begin{bmatrix} 0 & 1 & & \\ -100 & 0 & & \\ & & 0 & 1 \\ & & -225 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ .3 \\ 0 \\ .4 \end{bmatrix}$$

Transformation for A:

$$T = -\begin{bmatrix} 0 & 1/10 & 0 & 0 \\ 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/15 \\ 0 & 0 & 15 & 0 \end{bmatrix}$$

Rotationally-Destabilized Plant:

$$A_r = AT = \begin{bmatrix} -10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & -15 & 0 \\ 0 & 0 & 0 & 15 \end{bmatrix}$$

Note that half the eigenvalues of A_r are unstable. Optimal regulator design theory may now be used to stabilize the rotationally-destabilized plant.

DESIGN STEP 2: Stabilization of Rotational Plant

In this intermediate step, optimal regulator theory is applied to the plant in rotational space. The optimal gains that are generated will stabilize the rotationally-destabilized plant, but their prac-

tical function in this design algorithm is to structure a performance index that will rotate and stabilize an alpha-shift design for the *nominal* plant. The motivation for using the gains in this way came from applying them to the *nominal* plant and observing their significant effect: They eliminate harmonic components from the closed-loop system poles. State feedback with this set of optimal gains rotates the open-loop plant poles to the real-axis. This suggests that a SPI employing a feedback structure with these optimal gains as its state-weighting matrix can produce rotation and stabilization of a prescribed damping design.

An example of rotational stabilization is now presented for the low-order system used previously. The rotationally-destabilized plant matrix, A_r, developed in step 1, becomes a parametric design matrix in the algebraic Ricatti equation (ARE), i.e. the nominal plant, A, is replaced with A_r in the ARE. We note also that the parametric design matrix Q is set equal to zero in the example. This results in *double* poles in the closed-loop systems. Double poles are not mandatory. Alternative selections of Q may include an identity matrix which will split the closed-loop system poles, but still maintain them in the LHP. Positive scaling of the identity matrix will provide as much separation of the poles as desired. Negative scaling of the identity matrix Q adds harmonics to the closed-loop system and can be used, if desired, to obtain an additional increase in the modal frequencies in the final design step, or to decrease the optimal gains. Other structurings of the Q matrix are currently being evaluated for their closed-loop system effects. All examples in this paper employ a zero matrix Q which produces the double-pole structuring in the intermediate closed-loop systems. We now perform the rotational stabilization:

Algebraic Riccati Equation:

$$A^TP_1+P_1A-P_1BR^{-1}B^TP_1+Q=0$$

Parametric Design Matrices:

$$A=A_r$$
$$R=1$$
$$Q=[0]_{4x4}$$

Intermediate Riccati Solution for A_r:

$$P_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 5.5556E3 & 0 & -5.0000E3 \\ 0 & 0 & 0 & 0 \\ 0 & -5.0000E3 & 0 & 4.6875E3 \end{bmatrix}$$

Eigenvalues of Intermediate Closed-Loop Systems:

$$\lambda(A_r-BR^{-1}B^TP_1) = \{-10, -10, -15, -15\}$$
$$\lambda(A-BR^{-1}B^TP_1) = \{+5, +5, -30, -30\}$$

Note that P_1 is sparse and singular. Also, as indicated earlier, all harmonic components are completely eliminated from the closed-loop design model when the optimal gains are applied to the *nominal* plant, i.e. all poles have imaginary parts equal to zero. The real parts are positive or negative values which typically have values given by one-half or two times the modal frequencies. This intermediate closed-loop system must now be stabilized in a final design step with an additional algebraic Ricatti equa-tion which will also add a prescribed degree of harmonics to the closed-loop system.

DESIGN STEP 3: Prescribed Imaginary-Axis Pole Placement

In this section we develop the SPI that is employed to design optimal controllers and estimators

with prescribed (closed-loop) modal frequencies. The exponentially-weighted performance index of the alpha-shift technique is modified with the optimal gains of the rotational plant stabilization step. The modification of one SPI with the optimal gains from another results in a coupling of two *independently*-solved ARE's. As developed below, the coupling occurs through the ARE parametric matrix, Q.

The optimal gains generated in the rotational plant stabilization step are used to structure the parametric matrix, Q. "Q-equivalency", the expansion and collection of terms in an ARE to indirectly generate a state-weighting matrix, is used to structure Q. A Q_{eq} equation, parallel to that shown in Eq. 1, is developed for the rotationally-destabilized ARE in Eq. 2. (The unity subscripts indicate that this is the first ARE that is solved in the design algorithm.)

It is Q_{eq1} that is used to modify the exponentially-weighted performance index used in alpha-shift designs. The modified performance index and its accompanying ARE are shown in Eqs. 3a, 3b, and 4a respectively. Q_{eq1} transforms the alpha-shift design to rotational space. After rotation, the alpha parameter prescribes the amount of imaginary-axis pole translation that is desired from the *real-axis*.

A Q_{eq} may be developed for the modified ARE: Terms in Eq. 4a are expanded and collected as shown in Eq. 4b. Eq. 4c is formed by substitution of Q_{eq1} and defining $Q_{eq2} = 2\alpha IP_2$. Q_{eq} then is the sum of two terms--Q_{eq1} from the rotationally-destabilized ARE and Q_{eq2} from the alpha-shift design as shown in Eq. 5. If $Q_{eq2} >> Q_{eq1}$, i.e. if α is large relative to the modal frequencies, then the alpha-shift term will dominate, and the imaginary-parts of the poles will *asymptotically* approach the desired alpha value.

We now demonstrate prescribed imaginary-axis pole placement for the low-order system used previously. The optimal gains, P_1, designed under step 2 are used to form Q_{eq1} and modify the performance index for three alpha-shift designs: {α=0, α=1, and α=2}.

$$A_r^T P_1 + P_1 A_r - P_1 B R^{-1} B^T P_1 = 0$$

$$- P_1 B R^{-1} B^T P_1 + (A_r P_1 + P_1 A_r) = 0$$

$$Q_{eq1} = (A_r P_1 + P_1 A_r) = P_1 B R^{-1} B^T P_1 \qquad (2)$$

$$J_2 = \int_0^\infty e^{2\alpha\tau} [x^T Q_{eq1} x + u^T R u] \, dt \qquad (3a)$$

$$J_2 = \int_0^\infty e^{2\alpha\tau} [x^T P_1 B R^{-1} B^T P_1 x + u^T R u] \, dt \qquad (3b)$$

$$(A+\alpha I)^T P_2 + P_2 (A+\alpha I) - P_2 B R^{-1} B^T P_2 + P_1 B R^{-1} B^T P_1 = 0 \qquad (4a)$$

$$A^T P_2 + P_2 A - P_2 B R^{-1} B^T P_2 + P_1 B R^{-1} B^T P_1 + 2\alpha I P_2 = 0 \qquad (4b)$$

$$A^T P_2 + P_2 A - P_2 B R^{-1} B^T P_2 + Q_{eq1} + Q_{eq2} = 0 \qquad (4c)$$

$$Q_{eq} = Q_{eq1} + Q_{eq2} = P_1 B R^{-1} B^T P_1 + 2\alpha I P_2 \qquad (5)$$

Modified ARE: $(A+\alpha I)^T P_2 + P_2(A+\alpha I) - P_2 B R^{-1} B^T P_2 + P_1 B R^{-1} B^T P_1 = 0$

Q_{eq1} Formed From Optimal Gains P_1:

$$Q_{eq1} = P_1 B R^{-1} B^T P_1 = \begin{matrix} 0 & 0 & 0 & 0 \\ 0 & 1.1111E5 & 0 & -1.2500E5 \\ 0 & 0 & 0 & 0 \\ 0 & -1.2500E5 & 0 & 1.4062E5 \end{matrix}$$

Optimal Gains P_2 for Design 1: $\alpha=0$

6.2844E5	-5.1200E4	-1.0080E6	3.0400E4
-5.1200E4	9.2711E3	1.0440E5	-6.7200E3
-1.0080E6	1.0440E5	1.7601E6	-6.4800E4
3.0400E4	-6.7200E3	-6.4800E4	5.3025E3

Optimal Gains P_2 for Design 2: $\alpha=1$

7.1690E5	-5.2195E4	-1.1158E6	3.0532E4
-5.2195E4	1.2205E4	1.2129E5	-8.7124E3
-1.1158E6	1.2129E5	1.9794E6	-7.5562E4
3.0532E4	-8.7124E3	-7.5562E4	6.6674E3

Optimal Gains P_2 for Design 3: $\alpha=2$

8.2904E5	-4.8182E4	-1.2239E6	2.7139E4
-4.8182E4	1.6620E4	1.4013E5	-1.1753E4
-1.2239E6	1.4013E5	2.2091E6	-8.7720E4
2.7139E4	-1.1753E4	-8.7720E4	8.7729E3

Closed-Loop Eigenvalues $\lambda(A-BR^{-1}B^T P_2)$:

$\alpha=0$: {-5.0, -5.0, -30.0, -30.0}

$\alpha=1$: {-6.1 ± 1.0 i, -31.1 ± 1.0 i}

$\alpha=2$: {-7.2 ± 1.8 i, -32.2 ± 2.0 i}

For $\alpha=0$, the imaginary-parts of the closed-loop poles are zero rad/sec--stable, real-axis pole placement is achieved. The intermediate closed-loop system of step 2, characterized by $[A-BR^{-1}B^T P_1]$, has been stabilzed; only the RHP double poles at +5 are affected by the new optimal gain matrix, $R^{-1}B^T P_2]$.

For $\alpha=1$, the imaginary-parts of the closed-loop poles have values of 1 rad/sec. The real-parts of the eigenvalues have been increased from their values for the previous design as Q_{eq2} has begun to have an effect.

For $\alpha=2$, the imaginary-parts of the closed-loop poles have values

of 2 rad/sec., or approaching 2 rad/sec. As α increases, Q_{eq2} will begin to dominate the Q_{eq} term for the modified ARE--some closed-loop modal frequencies may be slightly less than the specified modal frequencies.

Step 3 concludes the prescribed imaginary-axis pole placement procedure for optimal controller design.

The optimal estimator is designed via duality theory using the same 3-step procedure.

We now present the design procedure in algorithmic form, and illustrate its effects on a higher-order example derived from flexible spacecraft control.

4. DESIGN ALGORITHM

Fig. 7 illustrates the design algorithm for prescribed imaginary-axis pole placement in the optimal controller system. Two independently-solved ARE's are employed: the ARE in rotational space, and the ARE in translational space. The coupling between the

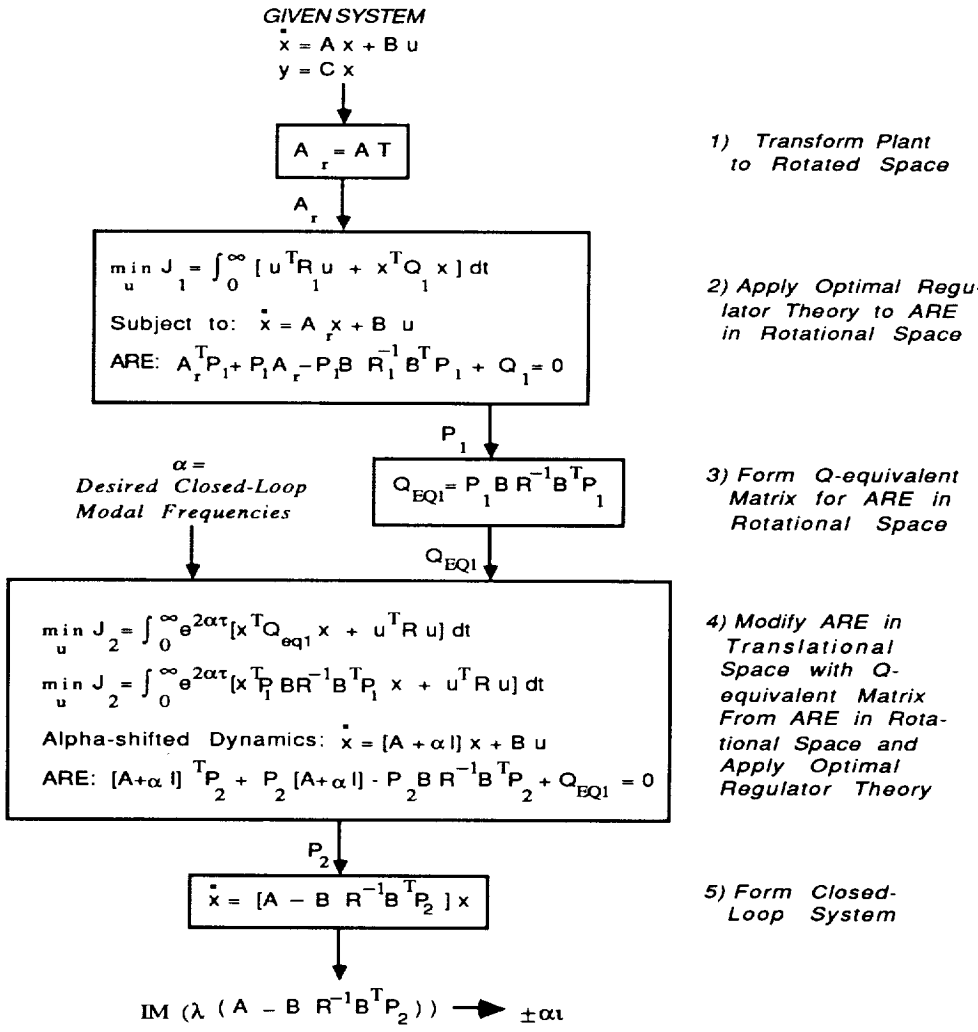


Figure 7. Design Algorithm

ARE's occurs with the Q-equivalent matrix for the ARE in rotational space. This Q-equivalent matrix acts as a state-weighting matrix for the ARE in translational space. The design algorithm for the optimal estimator follows a parallel structure: Dual variables are substituted into J_2, and A^T replaces A in the ARE in translational space.

5. FLEXIBLE SPACECRAFT CONTROL EXAMPLE

The design algorithm is applied to a model for the spacecraft boom shown in Fig. 8. The model contains twelve modes with frequencies ranging from 0.67 to 11.4 Hz. Four collocated actuators and sensors are positioned at the tip of the boom and mid-boom. All modes are modeled with zero damping.

For the example, we design five optimal controllers and compare their pole constellations. The

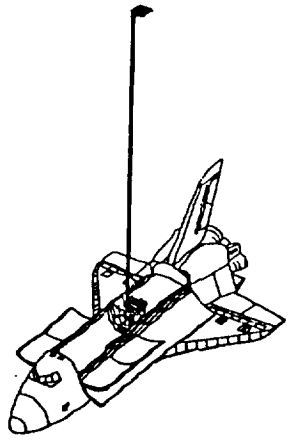


Figure 8. Flexible Spacecraft Boom

alpha values for the five designs, i.e. the prescribed imaginary-axis pole placement that is desired, are as follows: $\{\alpha_1=0, \alpha_2=1, \alpha_3=5, \alpha_5=10, \alpha_5=15\}$.

The design results are shown in Fig. 9 which plots the $\lambda(A-BR^{-1}B^TP_2)$ for the five designs. (Only the upper-half of the complex plane is shown.) Small values of

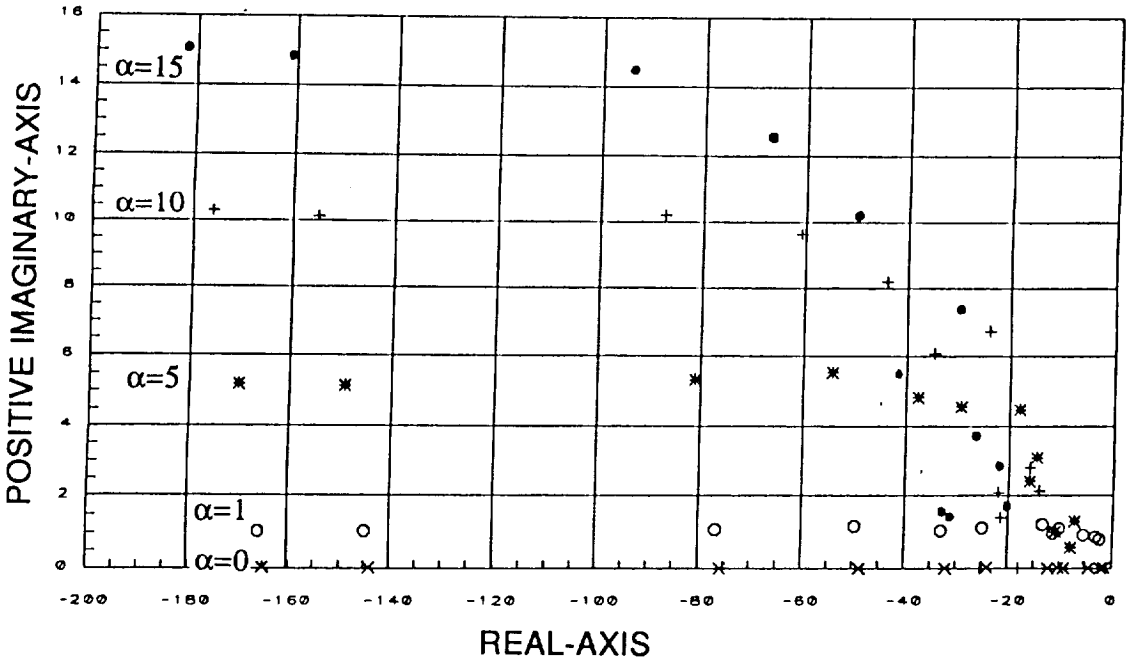


FIGURE 9. Controller Eigenvalues for Five Design Values of α

alpha (α_1=0, α_2=1) most closely approach the "prescribed" response, i.e. the imaginary-parts of all poles are approximately equal to the prescribed α value. This is due to a small or non-existent contribution of Q_{eq2} to the Q_{eq} matrix for the modified ARE, as explained in the previous section. For large values of α, the Q_{eq2} term begins to contribute to the Q_{eq} matrix, and the imaginary-parts of the pole asymptotically approach the desired α value. Poles corresponding to higher frequency modes have imaginary-parts that are closer to the α-asymptote.

6. SUMMARY/FUTURE WORK

A design procedure has been developed for prescribed-imaginary axis pole constraints for the optimal control and estimation systems: The imaginary-parts of the closed-loop system poles asymptotically approach a prescribed value, α. At this stage in the development, the maximum value that α may assume for a given system is constrained, possibly by a computational problem with solutions for the alpha-shifted ARE. Values of α that are large relative to the lowest modal frequency in the system can produce root migration from the desired α-asymptote. Small or mid-range frequency values of α produce excellent results as shown in the example of Section 5. Further analysis of the computational problem is required.

The design procedure developed empirically as the result of numerical experiments in gain transference and Q-equivalency theory. Future work calls for developing an analytical basis for the procedure. Additional work requires extending the design procedure to cover prescribed imaginary-axis constraints for the optimal compensator system.

BIBLIOGRAPHY

[1] B. D. O. Anderson and J. B. Moore, Linear Optimal Control, Prentice-Hall, 1971.

[2] J. R. Sesak and P. W. Likins, "Model Error Sensitivity Suppression: Quasi-Static Optimal Control for Flexible Structures," presented to 18th IEEE Conference on Decisin and Control, Fort Lauderdale, FL, Dec 1979.

[3] J. R. Sesak and K. M. Strong, "Robust Compensator Design Via Control and Observation Normalization," presented at AIAA Guidance, Navigation, and Control Conference, Boston, MA, 14-16 Aug, 1989.

Optimal Controllers for Finite Wordlength Implementation

K. Liu, R. Skelton

Purdue University

West Lafayette, IN 47907

ABSTRACT

When a controller is implemented in a digital computer, with A/D and D/A conversion, the numerical errors of the computation can drastically affect the performance of the control system. There exists realizations of a given controller transfer function yielding arbitrarily large effects from computational errors. Since, in general, there is no upper bound, it is important to have a systematic way of reducing these effects. Optimum controller designs are developed which take account of the digital round-off errors in the controller implementation and in the A/D and D/A converters. These results provide a natural extension to the LQG theory since they reduce to the standard LQG controller when infinite precision computation is used. But for finite precision the separation principle does not hold.

I. INTRODUCTION

LQG controllers are normally designed under the assumption that computer implemention will be perfect (this is the infinite wordlength assumption for state variable computation). However, real control systems are subject to the effects of finite wordlength computation. These round-off errors should not be ignored in the design of the controller. The influence of these errors on the control system and the optimum controller design considering their effects are the subjects of this paper.

We consider the problems that arise with fixed-point arithmetic and the finite word length of digital computers. This paper was motivated by the work of Kladiman and Williamson [1989]. Mullis and Roberts [1976] and Hwang [1977] in the field of signal processing first revealed the fact that the influence of round-off errors on digital filter performance depends on the realization chosen for the filter implementation. To minimize round-off errors these papers suggest a special coordinate transformation T prior to filter (or controller) synthesis.

This is in stark contrast to frequency domain approaches to control, which regard as irrelevant (and hence is completely ignored) the state space realization of the controller transfer function.

The idea of applying a coordinate transformation prior to controller synthesis has been applied to Kalman filter and LQG controller design problems, Williamson [1985], Kladiman and Williamson [1989]. One may select the wordlength of the computer to insure that the resulting degradation in the performance from round-off error is less than a certain percentage of the ideal behavior of the standard Kalman filter or LQG controller without round-off error. This approach was adapted by Sripad [1981] in the design of Kalman filters, and later by Moroney, et. al [1983] for LQG controller design. In these papers the standard Riccati equations are solved, *followed* by a coordinate transformation to reduce the effects of round-off errors. We shall call these controllers LQG_T to indicate a standard LQG controller followed by an "optimal" coordinate transformation T. This transformation *depends* on the control gains, hence, we put the word optimal above in quotes, because the standard LQG gain is not the optimal gain for the round-off

error problem. The optimum solution is to design the controller which *directly* takes into account the round-off errors associated with a finite word length implementation, rather than merely performing a coordinate transformation T on the LQG controller *after* it is designed. The optimal state estimation problem was solved by Williamson [1985]. This leads to a modified Kalman filter. The problem of optimum LQG controller design in the presence of round-off error was studied by Kadiman and Williamson [1989]. This paper worked with upper bounds and numerical results showed improvement over earlier work, but their algorithm does not provide the necessary conditions for an optimal solution. This paper provides the necessary conditions and a controller design algorithm for the solution of this problem. We shall call this controller LQG_{FW}.

With a fixed point implementation, the states of the LQG_{FW} controller are properly scaled to reduce the possibility of overflow. There are many scaling criteria available. The method we shall use is the variance oriented procedure, l_2-norm scaling [Hwang 1977]. We assume round-off errors are additive. This tends to be supported by the literature on state quantization, whereas quantization of coefficients leads to multiplicative errors [Williamson 1985].

The organization of the paper is as follows. In Section 2, the problem of LQG controller design in the presence of round-off errors is formulated. The importance of the coordinates of the controller will be discussed in Section 3. Section 3 summarizes the needed results from [Kadiman and Williamson 1989], and our new results on upperbounds of finite wordlength effects. It is shown that the portion of the LQG cost contributed by these errors will range from arbitrarily large to an achievable lower bound with the variation of the realization of the controller (variation of the choice of coordinates). The coordinate achieving the lower bound is described. In Section 4, the optimization problem is discussed in terms of chosing both the controller parameter matrices and the realization coordinate simultaneously. The necessary conditions are derived for the optimization problem. An algorithm is then presented for the designs of the optimal LQG_{FW} controller. The standard LQG and the LQG_{FW} controller are compared in Section 5. Some conclusions appear in Section 6.

II. Round-Off Error and LQG Controller Design Problem

In this section, we formulate the LQG controller design problem when round-off errors are present. The formulation procedure follows the original ideas of Mullis [1976], Hwang [1977] and the ideas of Williamson [1985], Kadiman and Williamson [1989]. Let us assume, for the study of round-off error, the discrete controller is designed from a discrete model of the plant to be controlled. We then introduce a model for finite wordlength effects into the discrete design problem.

Considering the following discrete-time model of a time-invariant plant:

$$\begin{cases} x_p(k+1) = A_p x_p(k) + B_p u(k) + D_p w_p(k) \\ z_p(k) = M_p x_p(k) + v_p(k) \\ y_p(k) = C_p x_p(k) \end{cases} \tag{1}$$

where x_p is the state n_p–vector, u, y_p and z_p are the control n_u–vector, output n_y–vector, measurement n_z–vector, v_p and w_p are assumed to be mutually independent, zero mean, discrete white Gaussian noises with covariance matrices V_p and W_p, respectively.

The controller that one might <u>desire</u> to implement is described by following equations:

$$\begin{cases} x_c(k+1) = A_c x_c(k) + B_c z_p(k) \\ u(k) = C_c x_c(k) + D_c z_p(k) \end{cases} \tag{2}$$

where x_c is the controller state n_c–vector, u and z_p are the control and measurement vectors described in the plant model. In a finite wordlength digital computer, the controller state x_c and measurement variable z_p will be quantized at each time of computation. Considering the quantization process, computation (1) and (2) cannot be accomplished. Instead the computation is described by

$$\begin{cases} x_p(k+1) = A_p x_p(k) + B_p Q[u(k)] + D_p w_p(k) \\ z_p(k) = M_p x_p(k) + v_p(k) \\ y_p(k) = C_p x_p(k) \end{cases} \tag{3a}$$

$$\begin{cases} x_c(k+1) = A_c Q[x_c(k)] + B_c\, Q[z_p(k)] \\ u(k) = C_c Q[x_c(k)] + D_c\, Q[z_p(k)] \end{cases} \tag{3b}$$

where $Q[\cdot]$ stands for the quantization process. Assuming an additive property of the round-off error, we can model the quantization process by:

$$Q[u(k)] = u(k) + e_u(k) \qquad \text{D/A} \tag{4a}$$

$$Q[x_c(k)] = x_c(k) + e_x(k) \qquad \text{control computer} \tag{4b}$$

$$Q[z_p(k)] = z_p(k) + e_z(k) \qquad \text{A/D} \tag{4c}$$

where e_u is the round-off error resulting from D/A conversion, $e_x(k)$ is the error resulting from quantization and $e_z(k)$ is the error resulting from A/D conversion. We do not claim that this assumption is always justified, but we invoke this common assumption in this paper, since one cannot *optimize* with respect to coefficient errors directly. One can only *evaluate* designs with respect to coefficient errors. There are many such evaluations in filter theory, and we shall add our own numerical evaluation in this paper. All such evidence points to a conclusion that controller structures that are good with respect to state quantization tend to also be good with respect to coefficient quantization.

It was shown [Sripad 1977] that, under sufficient excitation conditions, the round-off error $e_x(k)$ can be modeled as a zero mean, white noise independent of $w_p(k)$ and $v_p(k)$, with covariance matrix E_x,

$$E_x = qI, \quad q \triangleq \frac{1}{12} 2^{-2\beta} \ , \tag{5a}$$

where β is the wordlength of the control computer. Similarly, we assume the D/A conversion error $e_u(k)$ and the A/D conversion error $e_z(k)$ to be zero mean, mutually independent white

noise and also independent of $w_p(k)$, $v_p(k)$ and $e_x(k)$ with covariance matrices E_u and E_z,

$$E_u = q_u I, \quad q_u \triangleq \frac{1}{12} 2^{-2\beta_u} \tag{5b}$$

$$E_z = q_z I, \quad q_z \triangleq \frac{1}{12} 2^{-2\beta_z} \tag{5c}$$

where β_u and β_z are the wordlengths of D/A and A/D converters. Substitute (4) into (3) to obtain a closed-loop system model including finite wordlength effects,

$$\begin{cases} x_p(k+1) = A_p x_p(k) + B_p u(k) + D_p w_p(k) + B_p e_u(k) \\ z_p(k) = M_p x_p(k) + v_p(k) \\ y_p(k) = C_p x_p(k) \end{cases} \tag{6a}$$

$$\begin{cases} x_c(k+1) = A_c x_c(k) + B_c z_p(k) + A_c e_x(k) + B_c e_z(k) \\ u(k) = C_c x_c(k) + D_c z_p(k) + C_c e_x(k) + D_c e_z(k) \end{cases} \tag{6b}$$

We seek the controller to minimize the following cost function

$$J = \lim_{k \to \infty} E\{y_p^*(k) Q_p y_p(k) + u^*(k) R u(k)\} \tag{7}$$

where u and y_p are again control and output vectors, and Q_P and R are the weighting matrices.

After combining (6a) and (6b), and using the following notation for the vectors and matrices:

$$x(k) = \begin{bmatrix} x_p(k) \\ x_c(k) \end{bmatrix}; \; y(k) = \begin{bmatrix} y_p(k) \\ u(k) \end{bmatrix}; \; A = \begin{bmatrix} A_p & 0 \\ 0 & 0 \end{bmatrix}, \; B = \begin{bmatrix} B_p & 0 \\ 0 & I \end{bmatrix}, \; C = \begin{bmatrix} C_p & 0 \\ 0 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} D_p \\ 0 \end{bmatrix}; \; G = \begin{bmatrix} D_c & C_c \\ B_c & A_c \end{bmatrix}; \; I_0 = \begin{bmatrix} 0 & 0 \\ I & 0 \end{bmatrix}; \; I_1 = \begin{bmatrix} I \\ 0 \end{bmatrix}; \; I_2 = \begin{bmatrix} 0 \\ I \end{bmatrix};$$

$$M = \begin{bmatrix} M_p & 0 \\ 0 & I \end{bmatrix}, \; Q = \begin{bmatrix} Q_p & 0 \\ 0 & R \end{bmatrix}$$

the closed-loop system is compactly described by

$$x(k+1) = [A + BGM]x(k) + Dw_p(k) + BGI_1 v_p(k) + BGI_2 e_x(k) + BGI_1 e_z(k) + BI_1 e_u(k)$$
$$y(k) = [C + I_0GM]x(k) + I_0GI_1 v_p(k) + I_0GI_2 e_x(k) + I_0GI_1 e_z(k) \quad (9)$$

and the cost function (7) may be written

$$J = \lim_{k \to \infty} E\{y^*(k)Qy(k)\} . \quad (10)$$

Now, substitute (9) into (10), since $e_u(k)$, $e_x(k)$, $e_z(k)$, $w_p(k)$, and $v_p(k)$ are mutually independent,

$$J = tr\{X[C + I_0GM]^*Q[C + I_0GM]\} + tr\{V_p(I_0GI_1)^*Q(I_0GI_1)\}$$
$$+ tr\{E_x(I_0GI_2)^*QI_0GI_2)\} + tr\{E_z(I_oGI_1)^*Q(I_oGI_1)\} \quad (11a)$$

where X is the state covariance satisfying:

$$X = [A + BGM]X[A + BGM]^* + DW_pD^* + (BGI_1)V_p(BGI_1)^*$$
$$+ (BGI_2)E_x(BGI_2)^* + (BGI_1)E_z(BGI_1)^* + BI_1E_u(BI_1)^* \quad (11b)$$

We can decomposite J in eqn. (11a) into two terms:

$$J = J_{wv} + J_e \quad (12a)$$

where

$$J_{wv} \triangleq tr\{X_1[C + I_0GM]^*Q[C + I_0GM]\} + tr\{(V_p + E_z)(I_0GI_1)^*Q(I_0GI_1)\} \quad (12b)$$
$$X_1 = [A + BGM]X_1[A + BGM]^* + DW_pD^* + (BGI_1)(V_p + E_z)(BGI_1)^* + BI_1E_u(BI_1)^* \quad (12c)$$

and

$$J_e \triangleq tr\{X_e[C + I_0GM]^*Q[C + I_0GM]\} + tr\{E_x(I_0GI_2)^*Q(I_0GI_2)\} \quad (12d)$$
$$X_e = [A + BGM]X_e[A + BGM]^* + (BGI_2)E_x(BGI_2)^* \quad (12e)$$

where $X = X_1 + X_e$. J_{wv} is the portion of the performance index contributed by disturbances $e_u(k)$, $e_z(k)$, $w_p(k)$ and $v_p(k)$. J_e is the portion contributed solely by round-off error $e_x(k)$.

To prevent the overflow in controller state variable computation, we must properly scale the state variables. We use the l_2-norm scaling procedure which is written as:

$$[X_1(2, 2)]_{ii} = s \quad i = 1, \cdots, n_c \tag{13}$$

where $X_1(2, 2)$ is the (2.2) subblock matrix of X_1 matrix (the controller subblock), and $[\cdot]_{ii}$ stands for the ith diagonal element of the matrix. Equation (13) requires that the controller state variables have variance equal to s when the closed-loop system is excited only by outside disturbance and measurement noise. We call (13) the scaling constraint.

Therefore, the optimization problem is

$$\min_G J = \min_G (J_{wv} + J_e) \ , \tag{14}$$

subject to (12-13).

III. Contribution of Round-Off Error to the LQG Performance Index

In this section, we discuss the J_e term in (12a) and defined by eqn. (12d) which is the portion of the LQG cost function contributed by round-off errors. This portion of the cost function is coordinate dependent. It is unbounded from above, (that is, it can be arbitrarily large), but it has an achievable lower bound, which can be achieved in an optimal coordinate. The lower bound result was obtained by [Moroney et. al. 1983] and [Kadiman and Williamson 1989]. The construction of this optimal coordinate is discussed in this section, where we assume G is some given matrix (we shall optimize G later).

We will first present three key lemmas, which form the basis for the results of this section.

Lemma 1. [Mullis and Roberts 1976, Hwang 1977].

Given any $n \times n$ *matrix* M, *there exist a (non-unique) unitary matrix* U *such that* $(UMU^*)_{jj} = s$ *for all j, if and only if* $tr(M) = sn$

□

Lemma 2. [well known]

For any two positive definite matrices P *and* Q, *let* $\lambda_i[\cdot]$ *denote the* i^{th} *eigenvalue of matrix* $[\cdot]$. *Then,*

a) $\lambda_i[QP] > 0$ for all i

b) The $\lambda_i[QP]$ *are invariant under the transformation* $\tilde{P} = TPT^*$ *and* $\tilde{Q} = T^{-*}QT^{-1}$ *where* T *is nonsingular.*

□

Lemma 3.

Let a scalar J be defined by

$$J \triangleq \mathrm{tr}\{TT^*P\} \tag{15a}$$

where the $n_p \times n_p$ *nonsingular matrix* T *is* constrained by

$$(T^{-1}T^{-*})_{ii} = s \quad \text{for all i} \tag{15b}$$

and P *is a positive definite matrix. Then over the set of all nonsingular matrices* T *constrained by* (15b),

a) J *is not bounded from above.*

b) J *is bounded from below* ($J \geq \underline{J}$) *by*

$$\underline{J} \triangleq \frac{1}{sn_p}[\mathrm{tr}(\sqrt{P})]^2 \tag{16a}$$

where

$$P = \sqrt{P}\,\sqrt{P} \tag{16b}$$

and $\sqrt{P}$ *is symmetric.*

c) $\underline{J}$ *in* (16a) *is achievable by the matrix* T*:*

$$T = \underline{T} \triangleq U_t \Pi_t V_t^* \tag{17a}$$

where U_t, V_t *are unitary,* Π_t *diagonal, satisfying*

$$U_t \Pi_t^{-2} U_t^* = \frac{sn_p\sqrt{P}}{\mathrm{tr}(\sqrt{P})} \tag{17b}$$

$$[V_t \Pi_t^{-2} V_t^*]_{ii} = s \text{ for all i} \quad . \tag{17c}$$

□

Statements b) and c) are minor modifications of the results obtained by [Mullis and Roberts 1976] and [Hwang, 1977]. The proof of a) appears in Appendix A. An algorithm for solving (17b), (17c) is given in Appendix B.

The contribution of finite wordlength error in the cost function is described by equations (12d) and (12e). This J_e term can also be written as:

$$J_e = \mathrm{tr}\{K_e(BGI_2)E_x(BGI_2)^*\} + \mathrm{tr}\{E_x(I_0GI_2)^*Q(I_0GI_2)\} \tag{18a}$$

$$K_e = [A + BGM]^*K_e[A + BGM] + [C + I_0GM]^*Q[C + I_0GM] \ . \tag{18b}$$

Since $E_x = qI$, we then have:

$$J_e = q\,\mathrm{tr}\{(BGI_2)^*K_e(BGI_2) + (I_0GI_2)^*Q(I_0GI_2)\} \ . \tag{19}$$

We can easily check that the (2, 2)th subblock matrix of K_e (the controller subblock $K_e(2, 2)$) satisfies:

$$K_e(2, 2) = (BGI_2)^*K_e(BGI_2) + (I_0GI_2)^*Q(I_0GI_2) \ . \tag{20}$$

Substituting (20) into (19) reduces (19) to

$$J_e = q\mathrm{tr}[K_e(2, 2)] \ .$$

Hence, the minimization of J_e reduces to the problem:

$$\min J_e, \quad J_e = q\mathrm{tr}\{K_e(2, 2)\} \tag{21}$$

subject to (18b), (13) and (12c). From the singular value decompositions

$$X_1(2, 2) = U_x^*\Sigma_x U_x \tag{22a}$$

$$\Sigma_x^{1/2}U_xK_e(2, 2)U_x^*\Sigma_x^{1/2} = U_k^*\Sigma_k U_k \tag{22b}$$

then U_x, U_k are unitary, Σ_x, Σ_k are diagonal and

$$\Sigma_k \triangleq \text{diag}\ \{ \ldots\ \lambda_i[K_e(2,2)X_1(2,2)]\ \ldots\ \}\ . \tag{22c}$$

Suppose we begin our study with the closed-loop coordinate transformation T as:

$$T = \begin{bmatrix} I & 0 \\ 0 & U_x^* \Sigma_x^{1/2} U_k^* \end{bmatrix} . \tag{23}$$

Then, after this coordinate transformation as suggested by Kadiman and Williamson [1989]:

$$\bar{X}_1(2,2) = (U_x^* \Sigma_x^{1/2} U_k^*)^{-1} X_1(2,2) (U_x^* \Sigma_x^{1/2} U_k^{-*})^{-*} = I \tag{24}$$

$$\bar{K}_e(2,2) = (U_x^* \Sigma_x^{1/2} U_k^*)^* K_e(2,2) (U_x^* \Sigma_x^{1/2} U_k^*) = \Sigma_k\ . \tag{25}$$

If we take one more controller coordinate transformation T_c, the index J_e and its constraint equations, (after we substitute (24) and (25) into (13) and (21)), become

$$J_e = q\,\text{tr}[T_c T_c^* \Sigma_k] \tag{26a}$$

$$[T_c^{-1} T_c^{-*}]_{ii} = s, \quad i = 1, \cdots, n_c\ . \tag{26b}$$

Since, from Lemma 2, Σ_k in (22c) is coordinate independent, we may ignore the K_e and X_1 calculations (18b) and (12c) and concentrate on T_c in (26). Then, by applying Lemma 3 on equation (26), we have following theorem.

Theorem 1. *The round-off error term* J_e *in the LQG performance index* (12d) *and* (12e), *and constrained by the scaling constraint eqn.* (12c), (13), *is controller coordinate dependent. It is* unbounded from above *when the realization coordinate varies arbitrarily. It is* bounded from below *by the following lower bound:*

$$\underline{J}_e = \frac{q}{s n_c}\,\text{tr}\,\Sigma_k \tag{27}$$

The lower bound is achieved by the following controller coordinate transformation:

$$\underline{T}_c = U_x^* \Sigma_x^{1/2} U_k^* U_t \Pi_t V_t^* \tag{28a}$$

where U_x, U_k, U_t, V_t *are unitary matrices,* Σ_x, Π_t *are diagonal matrices, subject to the*

constraints:

$$X_1(2,\,2) = U_x^* \Sigma_x U_x \tag{28b}$$

$$\Sigma_x^{1/2} U_x K_e(2,\,2) U_x^* \Sigma_x^{1/2} = U_k^* \Sigma_k U_k \tag{28c}$$

$$U_t \Pi_t^{-2} U_t^* = \frac{sn_c\, \Sigma_k}{tr\Sigma_k} \tag{28d}$$

$$[V_t \Pi_t^{-2} V_t^*]_{ii} = s, \qquad i = 1, \cdots, n_c \quad . \tag{28e}$$

□

To find the optimal coordinate transformation $\underline{T}_c$ in (28a), we must solve (28d), (28e) to obtain U_t, Π_t, V_t. The equations (28d), (28e) are, however, special cases of (17b), (17c), where P is the diagonal matrix Σ_k. An algorithm is given in Appendix B to compute the U_t, Π_t, V_t needed for (28a).

The conclusion of this section is that the problem $\min_{T_c} J_e$ is solved by the coordinate transformation given by (28a).

IV. LQG Controller Design in the Presence of Round-Off Errors

As discussed in Section II, when round-off error is present, the LQG performance index can be decomposed into two terms. One term contains the influence of disturbance and measurement noise, the other term is contributed by round-off errors. Although the first term is not influenced by the coordinate of the controller, the second term is critically dependent on the coordinate. An optimal coordinate transformation is given by (28a). With the scaling requirement of the controller state variables to prevent overflow, we have a different optimization problem now for controller design comparing to the original optimal control design problem without round-off errors. In this section, we will discuss the controller design.

Let us first present a useful result.

Lemma 4. *Suppose* $J_{kx} \triangleq \sum_{l=1}^{n} \sqrt{\lambda_l[K(i,\, i)X(j,\, j)]}$ *where* $K(i,\, i)$ *and* $X(j,\, j)$ *are the* (i, i)th *subblock of* K *and* (j, j)th *subblock of* X *respectively. Define*

$$\nabla_k J_{kx} \triangleq \frac{\partial}{\partial K} J_{kx}\,, \qquad \nabla_x J_{kx} \triangleq \frac{\partial}{\partial X} J_{kx}$$

then:

a) $$\nabla_k J_{kx}(p,q) = 0 \qquad \textit{when } p \neq i \textit{ or } q \neq i \tag{29a}$$

$$\nabla_k J_{kx}(p,q) = \frac{1}{2}\sum_{l=1}^{n} \frac{[E^{-1}(i,\, j)]^{*}_{l\text{th-row}}[E(i,\, j)]^{*}_{l\text{th-col}} X(j,\, j)}{\sqrt{\lambda_l[K(i,\, i)X(j,\, j)]}} \quad \textit{when } p = i \textit{ and } q = i \tag{29b}$$

b) $$\nabla_x J_{kx}(p,\, q) = 0 \qquad \textit{when } p \neq j \textit{ or } q \neq j \tag{29c}$$

$$\nabla_x J_{kx}(p,\, q) = \frac{1}{2}\sum_{l=1}^{n} \frac{K(i,\, i)[E^{-1}(i,\, j)]^{*}_{l\text{th-row}}[E(i,\, j)]^{*}_{l\text{th-col}}}{\sqrt{\lambda_l[K(i,\, i)X(j,\, j)]}} \quad \textit{when } \; p = j \textit{ and } q = j \tag{29d}$$

where $\nabla_k J_{kx}(p,\, q)$ *and* $\nabla_x J_{kx}(p,\, q)$ *are the* (p, q)th *subblock of* $\nabla_k J_{kx}$ *and* $\nabla_x J_{kx}$, $E(i,\, j)$ *is the eigenvector matrix of matrix* $K(i,\, i)X(j,\, j)$

□

The proof of the lemma is given in Appendix A.

The LQG controller design problem, when finite wordlength effects are taking into account, are described by the equations (12-14). This is denoted as the LQG_{FW} controller. However, the scaling constraint (13) can be always satisfied by properly choosing the coordinates of the controller, so the problem breaks up into two parts: Finding G and finding its optimal coordinate transformation T_c to satisfy (12), (13) and (14). On the strength of Section 3, we can therefore write the optimization problem as

$$\min_{G,T_c} J = \min_{G,T_c} (J_{wv} + J_e) = \min_{G}[\min_{T_c} (J_{wv} + J_e)]$$

since J_{wv} is constant in terms of the variation of T_c, we have

$$\min_{G,T_c} J = \min_{G} [J_{wv} + \min_{T_c} J_e] \tag{30}$$

Assume $\underline{J}_e \triangleq \min_{T_c} J_e$ is given by (27), from Theorem 1. Hence, the equivalent LQG_{FW} design problem becomes

$$\min_{G}[J_{wv} + \underline{J}_e] \ , \tag{30a}$$

subject to (12c) and (18b) where

$$J_{wv} = \mathrm{tr} X_1 (C + I_0 GM)^* Q (C + I_0 GM) + \mathrm{tr}(V_p + E_z)(I_0 GI_1)^* Q (I_0 GI_1) \tag{30b}$$

$$\underline{J}_e = \frac{q}{sn_c} (\mathrm{tr}\, \Sigma_k)^2 \tag{30c}$$

where Σ_k is defined by (22c), and the transformation T_c which yields $\underline{J}_e$ is given by the algorithm in Appendix B, and may be computed only after the optimal G is obtained from (30). The following theorem states the necessary conditions of the optimization problem (30).

Theorem 2:

Necessary conditions for G *to be the solution of the optimal controller design problem* (30) *are:*

$$[A + BGM]X_1[A + BGM]^* + DW_pD^* + (BGI_1)(V_p + E_z)(BGI_1)^* + BI_1E_u(BI_1)^* - X_1 = 0 \quad (31a)$$

$$[A + BGM]^*K_e[A + BGM] + [C + I_0GM]^*Q[C + I_0GM] - K_e = 0 \quad (31b)$$

$$[A + BGM]^*K_2[A + BGM] + [C + I_0GM]^*Q[C + I_0GM] - K_2 + \nabla_x = 0 \quad (31c)$$

$$[A + BGM]K_3[A + BGM]^* - K_3 + \nabla_k = 0 \quad (31d)$$

$$(I_0^*QI_0 + B^*K_2B)G(MX_1M^* + I_1(V_p + E_z)I_1^*) + (I_0^*QI_0 + B^*K_eB)GMK_3M^* + \\ + B^*(K_2AX_1 + K_eAK_3)M^* = 0 \quad (31e)$$

where ∇_x *has 4 subblocks as*

$$\nabla_x(i, j) = 0 \qquad i \neq 2 \text{ or } j \neq 2$$

$$\nabla_x(2, 2) = \frac{q}{sn_c} \operatorname{tr} \Sigma_k \left\{ \sum_{i=1}^{n_c} \frac{K_e(2, 2)[E^{-1}]_{irow}^{-1*}[E]_{icol}^*}{\sqrt{\Sigma_{k_{ii}}}} \right\}$$

and ∇_k *also has* 4 *subblocks as*

$$\nabla_k(i, j) = 0 \qquad i \neq 2 \text{ or } j \neq 2$$

$$\nabla_k(2, 2) = \frac{q}{sn_c} \operatorname{tr} \Sigma_k \left\{ \sum_{i=1}^{n_c} \frac{[E^{-1}]_{irow}^*[E]_{icol}^* X_1(2, 2)}{\sqrt{\Sigma_{k_{ii}}}} \right\}$$

where E *is the matrix of eigenvectors of the matrix* $K_e(2,2)\, X_1(2,2)$.

□

The proof of theorem 2 is given in Appendix A.

Remark 1: The only terms in (31) which are affected by q are the two terms in (31c) and (31d) denoted by ∇_x, ∇_k. Hence setting $\beta = \infty$ gives $q = 0$, $\nabla_k = 0$, $\nabla_x = 0$, $K_3 = 0$, $K_2 = K_e$. Hence, eqs. (31) reduce to the standard LQG design by setting $\beta = \infty$. In this case, the 11 block of (31a) reduces to the Kalman filter Riccati equation, and the 22 block of (31c) reduces to the control Riccati equation.

Remark 2: We shall denote the controller satisfying (31) as the $\overline{LQG}_{FW}$ controller to indicate that the LQG_{FW} controller requires an additional step; the computation of $\underline{T}_c$ from Appendix B. Now, we have following LQG_{FW} controller design algorithm:

The LQG_{FW} Algorithm

Step 1: Solve G from equations (31a)-(31e). This gives the $\overline{LQG}_{FW}$ controller.

Step 2: Compute $\underline{T}_c = U_x^* \Sigma_x^{1/2} U_k^* U_t \Pi_t V_t^*$ by solving $U_x, \Sigma_x, U_k, U_t, \Pi_t, V_t$ from (28b)-(28e), using the G obtained in Step 1.

Step 3: $\tilde{G} = \begin{bmatrix} I & 0 \\ 0 & \underline{T}_c^{-1} \end{bmatrix} G \begin{bmatrix} I & 0 \\ 0 & \underline{T}_c \end{bmatrix}$ is the optimal LQG_{FW} controller for implementation.

□

Remark: A natural algorithm to suggest in Step 1 is as follows. Suppose one desires to design a LQG_{FW} controller for 10 bit arithmetic.

(i) Solve (31a)-(31e) for $\beta_i = \infty$, (hence, the standard LQG controller).

(ii) On the next iteration set β_i = 32 (or whatever gives a reasonably small number for ∇_x, ∇_k.

(iii) Iterate by indexing β_i. Change β_i by no more than one bit on each iteration. This gives an "answer" in 32-10 = 22 iterations (but this manner of choosing step sizes in not guaranteed to be sufficient to yield the optimal answer).

This is a "natural" homotopy method, since β is a natural choice for a homotopy parameter.

V. Computation Examples

We consider an Euler Bernoulli beam modeled by its first 5 bending modes with 2 inputs and 2 outputs. The modal frequencies appear in TABLE 1. In discrete controller design, the discrete model is represented by the matrices $\{A_p, B_p, C_p, D_p, M_p, W_p, V_p\}$ in equation (1). These matrices are given in Appendix C for a uniform sample time $\Delta t = 0.018$ sec. The LQG cost function is given by equation (7) with

$$Q_p = 0.99I \qquad R = 0.01I \ .$$

The wordlength of the control computer is assumed to be 4 bits. Since the effects of D/A and A/D conversion errors on the control system simply modify the effects of system disturbance and measurement noise, we ignore these errors in the example. Both the standard LQG controller and the LQG_{FW} controller are computed for the system.

	Frequency	Damping Factor
Mode 1	3.4987e+00	9.9994e-03
Mode 2	1.3995e+01	2.1301e-02
Mode 3	3.1488e+01	4.5600e-02
Mode 4	5.5979e+01	8.0400e-02
Mode 5	8.7468e+01	1.2530e-01

TABLE 1. Frequencies and Damping Factors of the Euler-Bernoulli Beam Example

The standard LQG controller of course was designed without consideration of round-off errors ($\beta = \infty$) and is labeled controller "LQG" in the TABLES. Controllers denoted "LQG_{Ti}" $i = 1, \cdots, 4$ are the same as the LQG, but for a coordinate transformation on the controller after G is computed. The matrices $\{A_c, B_c, C_c, D_c\}$ associated with the LQG_{T1} controller are shown in Appendix C. In different coordinates T_i, TABLE 2 shows the finite wordlength contribution J_e in the closed-loop system cost, using the standard LQG controller. In the optimal coordinate T_1 (controller LQG_{T1}) the cost J_e is about 500 times smaller than the cost in the original coordinate design (controller LQG). This improvement is equivalent to increasing the wordlength of the control computer by about 5 bits ($5 = \frac{1}{2} \log_2 500$). The effect of computational errors J_e in two commonly used coordinates, Normalized Observable Hessenberg Coordinates [Skelton 1988] and Phase Variable Coordinates, are also given in TABLE 2. The fact that Phase Variable Coordinates are bad for computation is consistent with other findings in filter synthesis [Williamson 1990]. The extreme high costs of the controller in a particular coordinate (LQG_{T4}) in TABLE 2 serves only to demonstrate that the cost J_e can become unbounded for some coordinates. The choice of coordinate T_4 was rather arbitrary and will not be described or discussed further.

Controller	Controller Coordinates	Cost J_e
LQG_{T1}	Optimal	9.793
LQG_{T2}	Normalized Obs. Hess.	2.692×10^2
LQG	Plant Coordinates	4.862×10^3
LQG_{T3}	Phase Variable	9.486×10^3
LQG_{T4}	Coordinate "X"	1.472×10^8

TABLE 2. Standard LQG Controller in Different Coordinates

The LQG_{FW} controller was designed by the LQG_{FW} algorithm given in Section 4. The controller matrices $\{A_c, B_c, C_c, D_c\}$ of this controller also appear in Appendix C. TABLE 3 shows the computed costs of the standard LQG controller, the transformed LQG controller (LQG_{T1}), and the LQG_{FW} controller (The "LQG_{FW} with coefficient error" will be discussed later). The costs for three different groups of excitations are computed in each case. The applicable disturbances for J, J_y, and J_u include plant disturbance w, sensor noise v, and finite wordlength error e. The applicable disturbance for J_e, J_{ey}, J_{eu} is only e, and for J_{wv}, J_{wvy}, J_{wvu} are only w_p and v_p (no finite wordlength effects). Hence, these sums apply to the various cost decompositions; J_y is the output term of J (the total cost), J_u is the control term in J, hence $J = J_y + J_u$. J_{wvy} is the output term of J_{wv} (the contribution of v_p and w_p in J), where $J_y = J_{wvy} + J_{ey}$ and $J_e = J_{ey} + J_{eu}$, $J = J_{wv} + J_e$. J_{wvu} is the control term of J_{wv} and $J_u = J_{wvu} + J_{eu}$. As we can

Disturbances Applied	Costs	LQG Controller	LQG_{T1} Controller	LQG_{FW} Controller	LQG_{FW} with coeff. errors
All v, w, and e	J	4.8827e+03	3.0589e+01	2.1207e+01	2.4695e+01
	J_y	2.8053e+03	2.3458e+01	2.0798e+01	2.4232e+01
	J_u	2.0774e+03	7.1303e+00	4.0941e-01	4.631e-01
e only	J_e	4.8621e+03	9.9302e+00	2.0067e-01	1.4071e-01
	J_{ey}	2.7850e+03	3.1790e+00	1.3841e-01	1.0275e-01
	J_{eu}	2.0771e+03	6.7512e+00	6.2267e-02	3.7961e-02
v and w only	J_{wv}	2.0659e+01	2.0659e+01	2.1006e+01	2.4554e+01
	J_{wvy}	2.0279e+01	2.0279e+01	2.0659e+01	2.0279e+01
	J_{wvu}	3.7912e-01	3.7912e-01	3.4715e-01	4.2514e-01

TABLE 3. Evaluation of LQG Controllers in Plant Coordinates, Optimal Coordinate and of the LQG_{FW} Controller

see in the TABLE 3, even when the standard LQG controller is in its optimal coordinate (LQG_{T1}), the J_e portion of the cost is still about 33% of the total cost (9.9302 compared to 30.589). By using the new LQG_{FW} controller design algorithm, we reduce the J_e portion of the cost 50 times, compared to the LQG_{T1} controller and 24,110 times compared to the LQG controller. In the latter case, this is equivalent to increasing the wordlength of the control computer by about 7 bits, That is, controller LQG_{FW} will give the same performance using 4 bit arithmetic that LQG gives using 11 bits. Furthermore this improvement in output performance is accompanied by a *reduction* in control effort RMS = $\sqrt{.40941}$ vs. RMS = $\sqrt{2077.4}$. To

conclude this point, we see that if both controllers use 4 bits, the difference in RMS output performance is an order of magnitude ($\sqrt{20.798}$ vs. $\sqrt{2805.3}$). This kind of improvement in performance can mean the difference between feasibility and infeasibility of some control missions.

With the new controller, the round-off portion J_e of the cost is only 0.85% of the total cost as opposed to 33% for LQG. Now let us discuss the cost J_{wv}, which would be the total cost if the closed-loop system was *only* excited by measurement noise v_p and disturbance w_p. That is, suppose the LQG_{FW} controller was *designed* for 4 bits, but *evaluated* using infinite bits. These are the conditions of the standard LQG design, since there are no disturbances *in the evaluation.* J_{wv} of the LQG_{FW} controller is a little higher than that of standard LQG controller. The output term of the cost is also a little higher and the control term a little lower. These indicate that the LQG_{FW} controller is a little more conservative than the designed standard LQG controller. This compromise in nominal performance allows robustness to computational errors. Note in TABLE 3, that the quantities that are optimized by the theory (under the given conditions) are shaded.

In the design of the LQG_{FW} controller, the equations (31a) to (31e) were solved iteratively by a gradient method. The standard LQG controller in its optimal coordinate (LQG_{T1}) was used as the initial controller design for starting the iterative process. Figs. 1-3 illustrate the convergence process for the LQG_{FW} algorithm, plotting the total cost J, the wordlength cost J_e, the the output J_y and input J_u performances, versus iteration. The optimal coordinate transformation played a crucial role in reducing the round-off errors (reducing the error by 3-4 orders of magnitude) as shown in Fig. 2. This was expected because the transformation was formulated in the optimization problem. The LQG_{FW} controller was obtained after about 300 iterative computations, but note from Figs. 1-3 that after 120 iterations one might have stopped with little loss.

Coefficient Errors

In the introduction we promised some evaluation of the effects of coefficient errors. We argued that even though the LQG_{FW} controller is optimized only for state quantization it performs well with coefficient quantization as well. To show this we introduced coefficient errors in the controller by using 4 bit precision instead of infinite precision in the controller coefficients. The key issue here is this. Quantization errors in the state degrades performance, but does not destabilize, since the effect of e is just a disturbance (note that all controllers in TABLEs 1 and 2 are stable). Coefficient errors can easily destabilize. Figure 4 shows the closed loop pole locations using the standard LQG regulator (using infinite precision). The system is stable as marked by the x's. When the controller coefficients are implemented using only 4 bit arithmatic, some poles as indicated by the o's in Fig. 4, are outside the unit circle. Hence the standard LQG controller is unstable using a 4 bit control computer.

Fig. 5 shows the improvement in the LQG controller by its optimal coordinate transformation before synthesis. This is the LQG_{T1} controller. The poles (o's) are in improved locations compared to Fig. 4, but the closed loop system is still unstable. The coordinate transformation helped but not enough. Fig. 6 shows the LQG_{FW} controller when controller coefficients are implemented using only 4 bits. The system is stable, confirming for this example improved robustness to controller coefficient errors, even though the controller has been optimized only for errors in controller state computation. The performance degradation in J, listed in the column "LQG_{FW} with coefficient errors" in TABLE 3 is about 15% (compared to nominal performance in TABLE 3).

Finally, we consider errors in *both* the plant and controller coefficients (due to quantization to 4 bits). These results are summarized in TABLE 4, where the modal damping in all modes is multiplied by parameter ρ. Hence $\rho=1$ corresponds to the nominal plant in all of the prior discussion. The range for stability using the LQG_{FW} controller is $.729 \leq \rho \leq 1.23$, demonstrating improved robustness over standard LQG controllers in the presence of errors in plant and controller coefficients.

Damping Error Factor ρ	LQG Controller	LQG_{T1} Controller	LQG_{FW} Controller
1.5242e+00	unstable	unstable	unstable
1.3717e+00	unstable	unstable	unstable
1.2346e+00	unstable	unstable	STABLE
1.1111e+00	unstable	unstable	STABLE
1.0000e+00	unstable (Fig 4)	unstable (Fig 5)	STABLE (Fig 6)
9.0000e-01	unstable	unstable	STABLE
8.1000e-01	unstable	unstable	STABLE
7.2900e-01	unstable	unstable	STABLE
6.5610e-01	unstable	unstable	unstable
5.9049e-01	unstable	unstable	unstable

TABLE 4. Robustness Controllers with respect to modal damping

(4-Bit Wordlength Controllers)

VI. Conclusion

This paper solves the problem of designing an LQG controller to be optimal in the presence of finite wordlength effects (modeled as white noise sources whose variances are a function of computer wordlength). This new controller, denoted LQG_{FW}, has two computational steps. First the gains are optimized, and then a special coordinate transformation must be applied to the controller. This transformation depends on the controller gains, so the transformation cannot be performed *a priori*. (Hence, there is no separation theorem.) The new LQG_{FW} controller design algorithm reduces to the standard LQG controller when an infinite wordlength is used for the controller synthesis, so this is a natural extension of the LQG theory. It was shown both theoretically and by example that the choice of controller coordinates significantly influences the effects of computational errors on the control system and that there exists an optimal set of coordinates in which to do these computations. Since we have not obtained a closed form solution for the LQG_{FW} problem, design of the LQG_{FW} controller by this algorithm requires significant computation. Hence, the improvement of the new controller is achieved at the expense of extra computational effort in design.

Acknowledgement: The importance of this problem was pointed out to us by Darrell Williamson. We gratefully acknowledge many helpful discussions with him, and the support of this work by NASA grant NAG1-857, Technical Monitor E.S. Armstrong.

Appendix A

1. Proof of Lemma 3

a) Using the singular value decomposition of $T = U_t \Pi_t V_t^*$, then the constraint equation (15b) becomes

$$(V_t \Pi_t^{-2} V_t^*)_{ii} = s \qquad \text{for all i} \tag{32}$$

from Lemma 1, above equation is equivalent to

$$tr(\Pi_t^{-2}) = sn_p \ . \tag{33}$$

Now, let us study the cost γ of (15a). Using the inequality

$$tr(AA^*) \geq \frac{[tr(AB^*)]^2}{tr(BB^*)}$$

we have a lower bound on γ

$$\gamma = \{U_t \Pi_t^2 U_t^* P\} = tr\{(\Pi_t U_t^* \sqrt{P})(\Pi_t U_t^* \sqrt{P})^*\}$$

$$\geq \frac{[tr\{(\Pi_t U_t^* \sqrt{P})(U_t^* [\sqrt{P}]^{-1})^*\}^2}{tr\{(U_t^* [\sqrt{P}]^{-1})(U_t^* [\sqrt{P}]^{-1})^*\}} = \frac{(tr\{\Pi_t\})^2}{tr\{P^{-1}\}} \tag{34}$$

Now, to prove that γ is unbounded from above, we prove that for any large scalar $m > 0$, we have $\gamma(\tilde{T}) \geq m$ for some $\tilde{T}$. Let us choose a $\tilde{T}$ having the following $\tilde{\Pi}_t$:

$$\tilde{\Pi}_t = \text{diag}\ (\tilde{\Pi}_i) \quad \text{such that}$$

$$\tilde{\Pi}_1 = \tilde{\Pi}_2 = \cdots = \tilde{\Pi}_{n_p - 2} = \frac{1}{\sqrt{s}}$$

and

$$\tilde{\Pi}_{n_p-1} = \frac{\sqrt{m\,tr(P^{-1})}}{\sqrt{2ms\,tr(P^{-1}) - 1}} \quad , \quad \tilde{\Pi}_{n_p} = \sqrt{m\,tr(P^{-1})}$$

where m is so chosen that

$$m > \frac{1}{2s\,tr(P^{-1})}$$

Then

$$tr(\tilde{\Pi}_t^{-2}) = \sum_{i=1}^{n_p} \frac{1}{\tilde{\Pi}_i^2} = s(n_p - 2) + \frac{2m\,tr(P^{-1}) - 1}{m\,tr(P^{-1})} + \frac{1}{m\,tr(P^{-1})} = sn_p \ .$$

Hence the chosen $\tilde{T}$ satisfies the constraint (33). Now, we have:

$$\gamma \geq \frac{(tr\{\tilde{\Pi}_t\})^2}{tr\{P^{-1}\}} = \frac{(\sum_{i=1}^{n_p} \tilde{\Pi}_i)^2}{tr\{P^{-1}\}} > \frac{(\tilde{\Pi}_{n_p})^2}{tr\{P^{-1}\}} = m$$

we then conclude the proof of part a). The proof of b) and c) follows next. The lower bound and the matrix T are found by using following inequality:

$$(trR)^2 \leq tr(QRQ^*)tr(Q^{-*}RQ^{-1}) \tag{35}$$

the equality holds above when $Q^*Q = \lambda^2 I$.

Let us assume $T = U_t \Pi_t V_t^*$, $P = U_p \Pi_p U_p^*$, where Π_t and Π_p are diagonal, U_t, V_t, U_p are unitary matrices. Assume for the R and Q matrices in (35),

$$R = U_t^* U_p \Pi_p^{1/2} U_p^* U_t \tag{36}$$

$$Q^*Q = U_t^* U_p \Pi_p^{1/4} U_p^* U_t \Pi_t^2 U_t^* U_p \Pi_p^{1/4} U_p^* U_t \ , \tag{37}$$

then

$$(Q^*Q)^{-1} = U_t^* U_p \Pi_p^{-1/4} U_p^* U_t \Pi_t^{-2} U_t^* U_p \Pi_p^{-1/4} U_p^* U_t \ .$$

Hence, we have:

$$\begin{aligned}
\mathrm{tr}(QRQ^*) &= \mathrm{tr}(RQ^*Q) = \mathrm{tr}[(U_t^*U_p\Pi_p^{1/2}U_p^*U_t)(U_t^*U_p\Pi_p^{1/4}U_p^*U_t\Pi_t^2U_t^*U_p\Pi_p^{1/4}U_p^*U_t)] \\
&= \mathrm{tr}[U_p\Pi_pU_p^*U_t\Pi_t^2U_t^*] = \mathrm{tr}[PTT^*] = \gamma \\
\mathrm{tr}(Q^{-*}RQ^{-1}) &= \mathrm{tr}[R(Q^*Q)^{-1}] = \mathrm{tr}[(U_t^*U_p\Pi_p^{1/2}U_p^*U_t)(U_t^*U_p\Pi_p^{-1/4}U_p^*U_t\Pi_t^{-2}U_t^*U_p\Pi_p^{-1/4}U_p^*U_t)] \\
&= \mathrm{tr}[U_p^*U_t\Pi_t^{-2}U_t^*U_p] = \mathrm{tr}[\Pi_t^{-2}]
\end{aligned}$$

From equation (33), and the above equation we have the following:

$$\mathrm{tr}(Q^{-*}RQ^{-1}) = \mathrm{tr}[\Pi_t^{-2}] = sn_p$$

Now, $\mathrm{tr}(R) = \mathrm{tr}(U_t^*U_p\Pi_p^{1/2}U_p^*U_t) = \mathrm{tr}(\Pi_p^{1/2}) = \mathrm{tr}(U_p\Pi_p^{1/2}U_p^*) = \mathrm{tr}\sqrt{P}$. Substitute the above equalities back into inequality (35). We then have: $[\mathrm{tr}(\sqrt{P})]^2 \le sn_p\gamma$, that is

$$\gamma \ge \frac{[\mathrm{tr}(\sqrt{P})]^2}{sn_p} \quad . \tag{38}$$

Now, suppose the matrix $\overline{T} = \overline{U}_t\overline{\Pi}_t\overline{V}_t^*$ yields the equality in (38). Since the equality in (35) holds when $Q^*Q = \lambda^2 I$, then we have:

$$\overline{U}_t^*U_p\Pi_p^{1/4}U_p^*\overline{U}_t\overline{\Pi}_t^2\overline{U}_t^*U_p\Pi_p^{1/4}U_p^*\overline{U}_t = \lambda^2 I \ ,$$

that is

$$\overline{U}_t\overline{\Pi}_t^2\overline{U}_t^* = \lambda^2U_p\Pi_p^{-1/2}U_p^* \Rightarrow \overline{U}_t\overline{\Pi}_t^{-2}\overline{U}_t^* = \lambda^2U_p\Pi_p^{1/2}U_p^* \ . \tag{39}$$

Hence

$$\overline{\Pi}^{-2} = \frac{\overline{U}_t^*U_p\Pi_p^{1/2}U_p^*\overline{U}_t}{\lambda^2} \ .$$

Substitute this $\overline{\Pi}^{-2}$ into equation (32) to obtain

$$(\overline{V}_t\overline{U}_t^*U_p\frac{\Pi_p^{1/2}}{\lambda^2}U_p^*\overline{U}_t\overline{V}_t^*)_{ii} = s \ .$$

Then $\mathrm{tr}\left[\frac{\Pi\sqrt{P}}{\lambda^2}\right] = sn_p$, hence $\lambda^2 = \frac{1}{sn_p}\mathrm{tr}(\Pi_p^{1/2}) = \frac{1}{sn_p}\mathrm{tr}(\sqrt{P})$. Now, substitute the above λ^2

into (39), to obtain

$$\bar{U}_t \bar{\Pi}_t^{-2} \bar{U}_t^* = \frac{sn_p U_p \Pi_p^{1/2} U_p^*}{tr(\sqrt{P})} = \frac{sn_p \sqrt{P}}{tr(\sqrt{P})} \tag{40}$$

Hence (38) yields the lower bound in (16a), and the matrix achieving this bound, shown by (40), must satisfy (17b). (17c) can be easily deduced from (15b). This concludes the proof.

□

2. Proof of Lemma 4

a) Proof of (29a): Since J_{kx} does not depend on K(p,q) for $p \neq i$ or $q \neq i$, we have:

$$\nabla_k J_{rx}(p, q) = \frac{\partial}{\partial K(p, q)} J_{k,x} = 0$$

Proof of (29b): We need following equality (e.g. Page 444 of Skelton [1988]) to prove the equation:

$$\lambda_i[A] = [E^{-1}]_{ith-row} A [E]_{ith-col}$$

where λ_i is the ith eigenvalue of A, and E the eigenvector matrix of A. Now, we have by taking A = K(i, i)X(j, j)

$$\lambda_l[K(i, i)X(j,j)] = [E^{-1}]_{lth-row} K(i, i)X(j, j)[E]_{lth-col}$$

$$= tr\{K(i, i)X(j, j)[E]_{lth-col}[E^{-1}]_{lth-row}\}$$

Hence from the differentiation rule $\frac{\partial trAB}{\partial B} = A^T$ we get

$$\frac{\partial \lambda_l}{\partial K(i,\, i)} = [E^{-1}]^T_{l\text{th-row}} [E]^T_{l\text{th-col}} X(j,\, j)$$

Then, we have:

$$\frac{\partial J_{k,x}}{\partial K(i,\, i)} = \frac{1}{2} \sum_{l=1}^{n} \frac{\dfrac{\partial}{\partial K(i,\, i)} \lambda_l [K(i,\, i) X(j,\, j)]}{\sqrt{\lambda_l [K(i,\, i) X(j,\, j)]}}$$

$$= \frac{1}{2} \sum_{l=1}^{n} \frac{[E^{-1}]^T_{l\text{th-row}} [E]^T_{l\text{th-col}} X(j,\, j)}{\sqrt{\lambda_l [K(i,\, i) X(j,\, j)]}} \,.$$

The proof of part b) follows in a similar manner

□

3. Proof of Theorem 2: Apply Lagrangian Multipliers K_2, K_3, then (30a)-(30c) leads to minimization of

$$\tilde{J} = \mathrm{tr}\{Q([C + I_0 GM] X_1 [C + I_o GM]^* + (I_0 GI_1)(V_p + E_z)(I_0 GI_1)^*)]\}$$

$$+ \mathrm{tr}\{K_2([A + BGM] X_1 [A + BGM]^* + DW_p D^* + (BGI_1)(V_p + E_z)(BGI_1)^* + (BI_1) E_u (BI_1)^*$$

$$- X_1)\} + \mathrm{tr}\{K_3([A + BGM]^* K_e [A + BGM] + [C + I_0 GM]^* Q$$

$$[C + I_0 GM] - K_e)\} + \frac{q}{sn_c} (\mathrm{tr}\Sigma_k)^2$$

Then

$$\frac{\partial \tilde{J}}{\partial K_2} = [A + BGM]X_1[A + BGM]^* + DW_pD^* + (BGI_1)(V_p + E_z)(BGI_1)^* + BI_1E_u(BI_1)^* - X_1 = 0$$

$$\frac{\partial \tilde{J}}{\partial K_3} = [A + BGM]^*K_e[A + BGM] + [C + I_0GM]^*Q[C + I_0GM] - K_e = 0$$

$$\frac{\partial \tilde{J}}{\partial X_1} = [C + I_0GM]^*Q[C + I_0GM] + [A + BGM]^*K_2[A + BGM] - K_2 + \nabla_{x_1} = 0$$

$$\frac{\partial \tilde{J}}{\partial K_e} = [A + BGM]K_3[A + BGM]^* - K_3 + \nabla_{k_1} = 0$$

Applying Lemma 4 on the above two equations, we can obtain ∇_{x_1} and ∇_{k_1} as stated in the theorem. This verifies (31a)-(31d). Now

$$\begin{aligned}\frac{\partial \tilde{J}}{\partial G} &= 2I_0^*QCX_1M^* + 2I_0^*QI_0GMX_1M^* + 2I_0^*QI_0GI_2(V_p + E_z)I_1^* + 2B^*K_2AX_1M^* \\ &+ 2B^*K_2BGMX_1M^* + 2B^*K_2BGI_1(V_p + E_z)I_1^* + 2B^*K_1AK_3M^* \\ &+ 2B^*K_1BGMK_3M^* + 2I_0^*QCK_3M^* + 2I_0^*QI_0GMK_3^*M^* = 0 \ ,\end{aligned}$$

but since $I_0^*QC = 0$, then,

$$\begin{aligned}\frac{\partial \tilde{J}}{\partial G} &= 2[I_0^*QI_0G(MX_1M^* + I_1(V_p + E_z)I_1^*) + B^*(K_2AX_1 + K_2AK_3)M^* \\ &+ (B^*K_1B + I_0^*QI_0)GMK_3M^* + B^*K_2BG(MX_1M^* + I_1(V_p + E_z)I_1^*)] \\ &= 2[(I_0^*QI_0 + B^*K_2B)G(MX_1M^* + I_1(V_p + E_z)I_1^*) + B^*(K_2AX_1 + K_1AK_3)M^* \\ &+ (B^*K_1B + I_0^*QI_0)GMK_3M^*] = 0 \ .\end{aligned}$$

This verifies (31e).

□

Appendix B

We now present an algorithm (originally developed by Hwang [1977]) for solving (17b) and (17c) for one set of solutions of U_t, Π_t, V_t (The solutions for U_t, Π_t, V_t are not unique). Let $\sqrt{P}$ in (17b) be written in terms of its singular value decomposition

$$\sqrt{P} = U_p \Sigma_p U_p^* \tag{41}$$

where U_p unitary, Σ_p diagonal.

Algorithm (Solving U_t, Π_t, V_t in (17b) and (17c))

I. Take:

$$U_t = U_p \tag{42a}$$

$$\Pi_t = \sqrt{\frac{tr(\Sigma_p)}{sn_p}\Sigma_p^{-1}} \tag{42b}$$

$$V_t = V_{n-1} V_{n-2} \cdots V_i \cdots V_2 V_1 \tag{42c}$$

where V_i, $i = 1, \cdots, n-1$ is computed as follows:

II. Compute V_1: Let

$$\Sigma_1 \triangleq \Pi_t^{-2} = \text{diag}\,(\cdots \sigma_{1j} \cdots) \tag{43a}$$

Assume σ_{11} and $\sigma_{1\beta}$ are two numbers such that one of them is bigger than s, the other is smaller than s. Then take V_1 as:

$$V_1 = \begin{array}{c} \\ \\ \\ \\ \beta \text{ row} \rightarrow \\ \\ \\ \\ \end{array} \begin{array}{c} \beta \text{ column} \\ \begin{bmatrix} f_1 & 0 & \cdots & 0 & g_1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & & \ddots & & \vdots & & & \vdots \\ 0 & & & 1 & 0 & & & \\ -g_1 & 0 & \cdots & 0 & f_1 & 0 & \cdots & 0 \\ 0 & & \cdots & \cdots & & 1 & & \vdots \\ \vdots & & & & & & \ddots & \\ 0 & & \cdots & \cdots & & \cdots & \cdots & 1 \end{bmatrix} \end{array} \tag{43b}$$

where

$$f_1 = \left[\frac{\sigma_{1\beta} - 1}{\sigma_{1\beta} - \sigma_{11}} \right]^{1/2} \tag{43c}$$

$$g_1 = \left[\frac{1 - \sigma_{11}}{\sigma_{1\beta} - \sigma_{11}} \right]^{1/2} \tag{43d}$$

Compute V_i: Let

$$\Sigma_i = V_{i-1}\, \Sigma_{i-1}\, V_{i-1}^* = \begin{bmatrix} \Sigma_{i1} & \Sigma_{i2} \\ \Sigma_{i2}^* & \Sigma_{i3} \end{bmatrix} \tag{44a}$$

where $\Sigma_i \in \mathbb{R}^{(i-1)x(i-1)}$ satisfies the property $[\Sigma_{i1}]_{jj} = s$, $\Sigma_{i2} \in \mathbb{R}^{(i-1)x(n-i+1)}$ is a nonzero matrix, and Σ_{i3} can be written as

$$\Sigma_{i3} = \begin{bmatrix} \sigma_{ii} & & 0 \\ & \ddots & \\ 0 & & \sigma_{nn} \end{bmatrix}$$

Assume σ_{ii} and $\sigma_{i\alpha}$ are numbers such that one of them is bigger than s and the other is smaller than s. Then take V_i as

$$
V_i = \begin{array}{c} \\ i\text{ row} \rightarrow \\ \\ \\ \alpha\text{ row} \rightarrow \\ \\ \end{array}
\begin{array}{c}
\begin{array}{ccccccccc} & & i\text{ column} & & & & \alpha\text{ column} & & \end{array} \\
\begin{bmatrix}
1 & & 0 & & & 0 & & \\
 & \ddots & \vdots & & 0 & \vdots & 0 & \\
0 & \cdots & f_i & 0 & \cdots & 0\ g_i & \cdots & 0 \\
 & & 0 & 1 & & 0 & & \\
 & 0 & \vdots & & \ddots & \vdots & 0 & \\
 & & 0 & & & 1\ 0 & & \\
0 & \cdots & g_i & 0 & \cdots & 0\ f_i & \cdots & 0 \\
 & 0 & \vdots & & 0 & \vdots & \ddots & \\
 & & 0 & & & 0 & & 1
\end{bmatrix}
\end{array}
\tag{44b}
$$

Compute f_i and g_i as:

$$f_i = \left[\frac{\sigma_{i\alpha} - 1}{\sigma_{i\alpha} - \sigma_{ii}} \right]^{1/2} \tag{44c}$$

$$g_i = \left[\frac{1 - \sigma_{ii}}{\sigma_{i\alpha} - \sigma_{ii}} \right]^{1/2} \tag{44d}$$

□

Computation of $\underline{T}_c$

$\underline{T}_c$ is formed as follows: $\underline{T}_c \triangleq U_x^* \Sigma_x^{1/2} U_K^* U_t \Pi_t V_t^*$

1) Compute the Covariance Matrix and Observability Grammian

$$K_e = [A + BGM]^* K_e [A + BGM] + [C + I_o GM]^* Q [C + I_o GM]$$

$$X_1 = [A + BGM] X_1 [A + BGM]^* + DW_p D^* + (BGI_1)(V_p + E_z)(BGI_1)^* + BI_1 E_u BI_1$$

Assume $K_e(2,2)$, $X_1(2,2)$ to be (2,2) the subblocks of K_e and X_1 (the controller subblocks).

2) Compute U_x, Σ_x, U_k.

These three matrices are computed by applying singular value decomposition on following matrices:

$$X_1(2,2) = U_x^* \Sigma_x U_x$$

$$\Sigma_x^{1/2} U_x K_2(2,2) U_x^* \Sigma_x^{1/2} = U_k^* \Sigma_k U_k$$

3) Compute U_t, Π_t, V_t.

Let us replace P matrix in the algorithm of appendix B as

$$P \triangleq \text{diag}\,[\lambda_i\{K_e(2,2)X_1(2,2)\}]$$

Then we can compute U_t, Π_t, V_t by applying the algorithm on matrix P.

Appendix C

DESIGN EXAMPLE OF ROUND-OFF LQG CONTROLLER

==========

Plant Model: 10th Order Euler-Bernoulli Beam

Word-Length of the Assumed Computer: 4 bits

1) The 10th Order Euler-Bernoulli Beam Model for Controller Design

$$A = \begin{bmatrix} 0.9980 & 0.0179 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.2196 & 0.9968 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9687 & 0.0177 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -3.4620 & 0.9582 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.8469 & 0.0166 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -16.4457 & 0.7993 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5594 & 0.0139 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -43.6477 & 0.4340 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1138 & 0.0095 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -72.4045 & 0.0937 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0014 & 0.0006 \\ 0.1557 & 0.0716 \\ -0.0004 & 0.0011 \\ -0.0480 & 0.1257 \\ -0.0012 & 0.0013 \\ -0.1299 & 0.1440 \\ 0.0007 & 0.0012 \\ 0.0720 & 0.1164 \\ 0.0007 & 0.0007 \\ 0.0588 & 0.0588 \end{bmatrix} \qquad D = \begin{bmatrix} 0.0014 & 0.0006 \\ 0.1557 & 0.0716 \\ -0.0004 & 0.0011 \\ -0.0480 & 0.1257 \\ -0.0012 & 0.0013 \\ -0.1299 & 0.1440 \\ 0.0007 & 0.0012 \\ 0.0720 & 0.1164 \\ 0.0007 & 0.0007 \\ 0.0588 & 0.0588 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 7.8297 & 0 & 7.1091 & 0 & -1.3744 & 0 & -8.3569 & 0 & -6.2128 \\ 0 & 6.2128 & 0 & -8.7875 & 0 & 6.2128 & 0 & 0 & 0 & -6.2128 \end{bmatrix}$$

$$M = \begin{bmatrix} 0 & 7.8297 & 0 & 7.1091 & 0 & -1.3744 & 0 & -8.3569 & 0 \\ 0 & 6.2128 & 0 & -8.7875 & 0 & 6.2128 & 0 & 0 & 0 \end{bmatrix}$$

$$W = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad V = \begin{bmatrix} 1.0003e{-}03 & 0 \\ 0 & 1.0003e{-}03 \end{bmatrix}$$

2) Designed Regular LQG Controller in Optimal Coordinate LQG_{T1}

$$A_c = \begin{bmatrix} -0.4582 & -0.1633 & -0.0133 & -0.1836 & 0.1574 & -0.4386 & -0.1054 & -0.2805 & 0.2304 & -0.2815 \\ 0.4144 & 0.6040 & 0.4587 & -0.4122 & -0.0201 & -0.0411 & 0.2748 & 0.1059 & -0.0786 & 0.0379 \\ 0.0849 & -0.5217 & 0.5622 & -0.3257 & 0.3373 & 0.2351 & 0.0665 & 0.1975 & -0.1651 & 0.2658 \\ 0.4753 & -0.3503 & 0.2226 & 0.5105 & -0.3084 & 0.0821 & 0.4446 & 0.1978 & -0.1382 & -0.0456 \\ 0.3326 & 0.0383 & -0.5299 & -0.1864 & 0.4324 & 0.3391 & 0.3306 & 0.2351 & -0.1635 & -0.1155 \\ 0.2946 & -0.1855 & -0.0.850 & -0.3095 & -0.2941 & -0.0605 & -0.7404 & 0.0085 & 0.1530 & 0.5389 \\ 1.5034 & -0.2726 & -0.0095 & -0.2270 & -0.0416 & -0.4845 & -1.5704 & -0.3867 & -0.0236 & -0.4084 \\ 0.5293 & 0.0908 & -0.0359 & -0.0617 & -0.3343 & -0.0787 & -1.0273 & -0.1971 & -0.0491 & 0.4129 \\ -0.0468 & -0.0574 & -0.0709 & -0.0716 & -0.0416 & 0.1318 & 0.5827 & -0.9215 & -0.0746 & 0.2806 \\ -0.4312 & 0.1539 & -0.0256 & 0.0559 & -0.1463 & 0.4745 & -0.0777 & -0.3449 & -0.9854 & -0.6735 \end{bmatrix}$$

$$B_c = \begin{bmatrix} 0.1894 & -0.2895 \\ -0.422 & 0.0230 \\ -0.0296 & 0.0941 \\ -0.0120 & -0.0024 \\ -0.0258 & 0.0940 \\ -0.0611 & 0.0609 \\ -0.2200 & 0.4919 \\ -0.0737 & 0.2522 \\ 0.0252 & -0.0076 \\ 0.0737 & -0.0776 \end{bmatrix} \qquad D_c = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$C_c = \begin{bmatrix} -1.9370 & 3.8601 & 4.1659 & 3.4458 & 1.8923 & -4.2436 & -15.7358 & -6.5380 & 3.7048 & -5.3330 \\ 1.6850 & -3.2381 & -2.7357 & -2.8624 & -2.7744 & 5.2406 & 11.5365 & 3.9625 & -2.8745 & -0.1711 \end{bmatrix}$$

3) LQG_{FW} Controller from the LQG_{FW} Algorithm of Section 4

$$A_c = \begin{bmatrix} 0.3501 & 0.4306 & -0.2223 & 0.3078 & -0.5350 & 0.1231 & 0.1595 & -0.2003 & -0.1024 & 0.1325 \\ -0.2004 & -0.2851 & -0.2294 & 0.1810 & 0.6715 & -0.4432 & 0.2756 & 0.1591 & 0.3525 & -0.3974 \\ -0.2033 & 0.2556 & -0.0197 & -0.8326 & -0.8293 & 0.0885 & -0.0605 & 0.2870 & -0.0571 & 0.1147 \\ -0.2973 & -0.3621 & 0.6480 & 0.3770 & -0.4095 & 0.4031 & -0.2736 & 0.0125 & 0.0426 & -0.0372 \\ 0.0308 & -0.2207 & -0.5168 & -0.3001 & -0.9847 & 1.1705 & -1.0703 & 0.7456 & -0.0979 & 0.1920 \\ -0.1187 & 0.4836 & 0.0470 & 0.3655 & 0.2493 & -1.0109 & 0.3516 & 0.5930 & 0.2744 & -0.3872 \\ 0.0089 & -0.3363 & 0.0664 & -0.0869 & 0.0085 & -0.0712 & -0.1936 & 0.113 & 0.3818 & -0.5248 \\ -0.3341 & 0.2935 & 0.1055 & 0.1309 & 0.2251 & -0.3631 & -0.7912 & -0.5655 & -0.2610 & 0.3180 \\ 0.0731 & -0.0312 & -0.0788 & -0.1349 & -0.4369 & 0.2594 & -0.4096 & -0.3895 & 0.7609 & 0.3237 \\ -0.1129 & 0.0070 & 0.0781 & 0.1679 & 0.4955 & -0.3354 & 0.5865 & 0.4685 & 0.2460 & 0.6396 \end{bmatrix}$$

$$B_c = \begin{bmatrix} -0.0134 & 0.0927 \\ 0.0812 & -0.1630 \\ 0.0706 & -0.3987 \\ 0.2464 & -0.6411 \\ 0.4583 & -1.0134 \\ -0.5942 & 1.0745 \\ 0.2455 & -0.2146 \\ 0.1121 & 0.0815 \\ 0.1013 & -0.2475 \\ -0.1510 & 0.3465 \end{bmatrix} \qquad D_c = \begin{bmatrix} -0.4486e{-}04 & -0.1328e{-}04 \\ -0.5913e{-}04 & -0.1567e{-}04 \end{bmatrix}$$

$$C_c = \begin{bmatrix} 0.8861 & -1.8997 & 3.8592 & -0.3107 & 5.3072 & -0.7395 & 0.6339 & -1.6517 & 0.9202 & -1.3734 \\ -1.4019 & 2.2532 & -2.6576 & 0.1575 & -3.3358 & 1.2007 & -0.5179 & 0.2062 & -0.9969 & 1.3884 \end{bmatrix}$$

References

[1] S. Hwang [1977]; "Minimum Uncorrelated Unit Noise in State-Space Diginal Filtering;" IEEE Trans, Acoust. Speech, Signal Processing; Vol-25; Aug. 1977; pp. 273-281.

[2] K. Kadiman and Williamson [1989]; "Optimal Finite Wordlength Linear Quadratic Regulation" IEEE Trans. on Automatic Contr., Vol. 34, No. 12, pp. 1218-1228, Dec. 1989.

[3] H. Kwakernaak; R. Sivan [1972]; "Linear Optimal Control Systems;" John Wiley & Sons.

[4] P. Lancaster [1969]; "The Theory of Matrix;" Academic Press.

[5] D. Luenberger [1984]; "Linear and Nonlinear Programming;" Addison-Wesley.

[6] P. Moroney; A. Willsley; P. Houpt [1983]; "Round-Off Noise and Scaling in the Digital Implementation of Control Compensators;" IEEE Trans; Acoust. Speech, Signal Processing; Vol-31; Dec. 1983; pp. 1464-1477.

[7] C. Mullis and R. Roberts [1976]; "Synthesis of Minimum Round-Off Noise Fixed Point Digital Fiters;" IEEE Trans.; Circuits and Syst.; Vol-23; Sept. 1976; pp. 551-562.

[8] R. Skelton [1988]; "Dynamical System Control;" John Wiley & Son.

[9] A. Sripad; D. Synder [1977]; "A Necessary and Sufficient Condition for Quantization Error to be Uniform and White;" IEEE Trans. Acout. Speech, Signal Processing; Vol-5; Oct. 1977; pp. 442-448.

[10] A. Sripad [1981]; "Performance Degradation in Digitally Implemented Kalman Filter;" IEEE Trans. Aerospace Electron. System; Vol-17; Sept. 1981; pp. 626-634.

[11] D. Williamson [1985]; "Finite Word Length Design of Digital Kalman Filters for State Estimation;" IEEE Trans. on Automatic Contr.; Vol-30; Oct. 1985; pp. 30-39.

SYSTEM DYNAMIC SIMULATION OF PRECISION SEGMENTED REFLECTOR

Choon-Foo Shih and Michael C. Lou
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California 91107

4th NASA Workshop on
Computational Control of Flexible Aerospace Systems
July 11-13, 1990

INTRODUCTION

To develop enabling technologies needed for future advanced astrophysics missions, two NASA centers, the Jet Propulsion Laboratory (JPL) and the Langley Research Center (LaRC), are undertaking a joint effort on a Precision Segmented Reflector (PSR) Project. The missions to which PSR is intended to support include the Submillimeter Explorer (SMME) and Submillimeter Infrared Line Survey (SMILS), both planned for the mid-1990's, and the Large Deployable Reflector (LDR) for the early 2000's. All of these mission will employ large (up to 20 meters in diameter) telescopes. The essential requirement for the telescopes is that the reflective surface of the primary mirror must be made extremely precise to allow no more than a few microns of errors and, additionally, this high surface precision must be maintained when the telescope is subjected to on-orbit mechanical and thermal disturbances. Based on the mass, size, and stability considerations, reflector surface formed by segmented, probably actively or passively controlled, composite panels are regarded as most suitable for future space-based astronomical telescope applications.

In addition to the design and fabrication of composite panels with a surface error of less than 3 microns RMS, PSR also develops related reflector structures, materials, control, and sensing technologies. Furthermore, a Technology Demonstration has been proposed to illustrate hardware integration, study interaction of technologies, and evaluate system performance. As part of the planning effort for PSR Technology Demonstration, a system model which couples the reflector, consisting of panels, support truss and actuators, and the optical bench was assembled for dynamic simulations. Random vibration analyses using seismic data obtained from actual measurements at the test site designated for PSR Technology Demonstration are described in this paper.

BACKGROUND

- **The Precision Segment Reflector (PSR) Program was initiated in early 1988 as an element of NASA's Civilian Space Technology Initiative (CSTI).**
- **A joint LaRC/JPL effort.**
- **To develop enabling technologies needed for future astrophysics missions**
  - **Large Deployable Reflector (LDR)**
  - **Submillimeter Explorer (SMME), Submillimeter Infrared Line Survey (SMILS)**
- **Four major elements are included in the PSR technology development**
  - **Lightweight composite panels**
  - **Lightweight support structures**
  - **Panel figure control**
  - **System technology demonstration**

STRUCTURAL CONFIGURATION

The current baseline LDR telescope system, illustrated in the sketch shown in Figure 1, has a 20-meter filled aperture reflector with the reflective surface form by five rings of 84 hexagon-shaped, lightweight, composite panels[1]. The backup structure employed to support these panels is a tetrahedral, space-erectable truss constructed with thin-walled composite struts. In order to conduct astronomical observations in the sub-millimeter/far-infrared wavelength range of 30 to 50 microns, the LDR is required to have a surface precision that allows no more than a few microns (root-mean-square) errors.

As a precursor technology development effort for the LDR-class space optical systems, the Precision Segmented Reflector (PSR) Program was initiated in 1988 as one of the major elements of NASA's Civil Space Technology Initiative (CSTI). The PSR (Figure 2) has a parabolic reflective surface that is formed by 19 hexagonal composite panels and with a focal length of 2.4 meters. The nominal size of each PSR hexagonal composite panel is 0.9 meters, measured from vertex to vertex. When fully assembled, all PSR panels except the central one will be actively controlled by voice-coil actuators. There will be three actuators for each panel to accomplish controlled motions for three degrees of freedom, one piston and two tilts.

In the PSR structures area, the major accomplishment has been the successful development of the PSR Testbed (TB) truss structure [2]. This space-erectable truss structure, consisting of 45 aluminum nodes, 300 aluminum joints and 150 graphite-epoxy composite struts, was designed, analyzed, fabricated, and assembled at LaRC. Photogrammetry survey performed on the as-assembled PSR TB truss structure indicated that the RMS error of positioning accuracy for the 27 upper surface nodes is about 70 microns and is substantially better than the 100 microns goal. Structural tests including static deflection and modal survey were also conducted and correlated with analytical predictions [3].

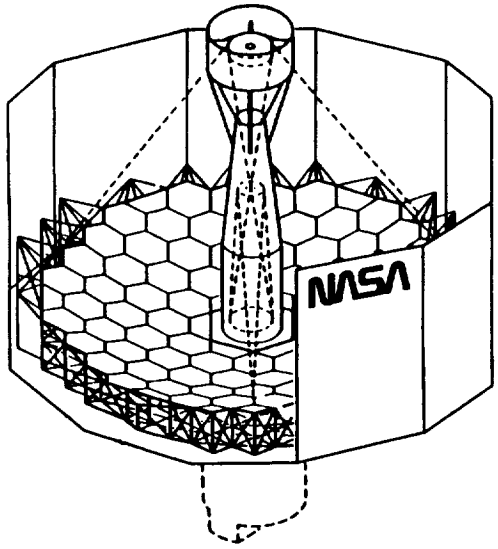


Fig.1 Large Deployable Reflector (LDR)

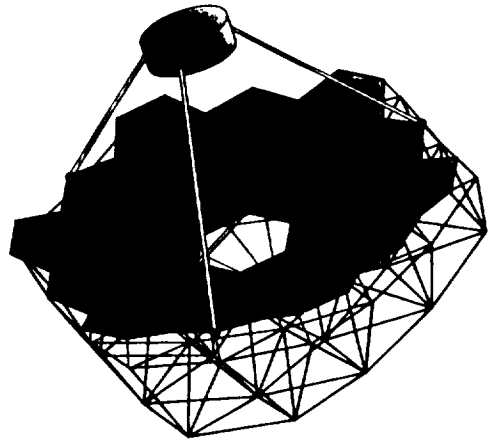


Fig.2 Precision Segmented Reflector (PSR)

ARTICULATED PANEL MODULE (APM)

Another significant accomplishment related to the PSR structural effort is the development of the Articulated Panel Module (APM) design concept for attaching panels to the support truss. The APM is a modular design specifically developed to provide well-defined, "soft-support" interface between the PSR composite panels and the TB truss structure (Figure 3). It also provides physical support to the control actuators and serves as the optical bench for the edge sensors employed for aligning neighboring panels.

Specific PSR/APM design requirements for the 0.9 m panel are described in Reference 4. The allowable panel movements and panel offset are applied to define the geometries of the APM components. The flexure sizes, as well as the dimensions of the lateral constraint struts, are derived from the specifications of the desired natural frequency range. The current APM configuration has been designed so that the natural frequencies of the piston mode and the tilt modes are less than 0.2 Hz and the natural frequencies of the rotational and the lateral modes are somewhat near 50 Hz. In addition, the non-rigid spatial deformation of the front panel facesheet above the interface node is not allowed to exceed 20 nm. over a 6.6 cm. distance with a temperature difference of 2°C. This thermal deformation requirement led us to choose INVAR as the panel interface fitting material.

Various design considerations and solutions had to be addressed in the design of a prototype APM that would accommodate all the functional requirements and the design criteria. The first design consideration was to establish low thermal expansion coefficients in the overall APM components for an expected 200 K space operational environment. This CTE consideration was solved by using low CTE materials through the entire APM. The proposed materials are graphite/epoxy, titanium and INVAR-36. The consideration of design simplicity was met through the proper design configuration. There are only three panel interface points in the current APM design. The lateral constraint struts were placed inside the subframe tubes in order to reduce the packaging complexity. The lightweight consideration was fulfilled by choosing lightweight materials. That is why graphite/epoxy was used for the lateral constraint struts and the subframe tubes, moreover, titanium was proposed for all the fittings and flexures. Fittings are applied in order to facilitate the APM assembly. Flexures are used in the APM for both precise and predictable considerations. A description of the APM development, including details on its structural and functional requirements and design approaches, is presented in Reference 5.

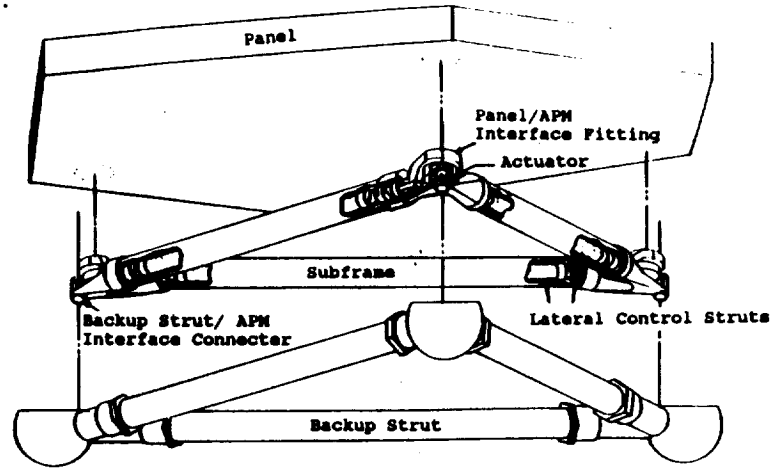


Fig.3 APM,Panel,and Backup Struts

PSR TECHNOLOGY DEMONSTRATION MODEL

The PSR TD is a test and demonstration effort with the following specific objectives: (1) demonstrate the integration of panels, backup structure, APM and figure sensing hardware components developed within the PSR program; (2) validation of individual PSR component technologies in a complete telescope reflector system environment; (3) development of ground test methods for large precision space structures; and (4) generation of experimental data for comparison with results predicted by an optical performance simulation model. Figure 4 is one of the baseline test configuration proposed for the PSR TD. Only one of the nineteen composite panels will be actively controlled in the PSR TD tests. The actively controlled panel can be located on either the first (inner) or the second (outer) ring of the reflective surface, however, the final locations for actively controlled panels have not been selected.

The structural model of the PSR TD system includes the panels, the APM and the backup support truss. However, the optical bench is not included in the PSR TD system model. This is because of that the structural design of the optical bench has not been completed and its stiffness is considered to be relatively rigid compared to the TD structural system. The panels incorporated in the PSR TD program are a hexagonal shape and of a 2-inch thick aluminum core and 0.04-inch thick composite facesheets. The corresponding lowest natural frequency of the panel itself is about 200 Hz[6]. Two PSR TD structural models were assembled in the present study. The first model (System I) is based on the assumption that the actively-controlled panel is attached to the first ring of the backup support struts, as shown in Figure 5.a. The System II model assumes that the actively-controlled panel is attached at the second ring of the backup struts, as shown in Figure 5.b. The boundary conditions of both systems are assumed to be rigidly mounted to the ground at the three inner nodes of the lower surface of the backup truss.

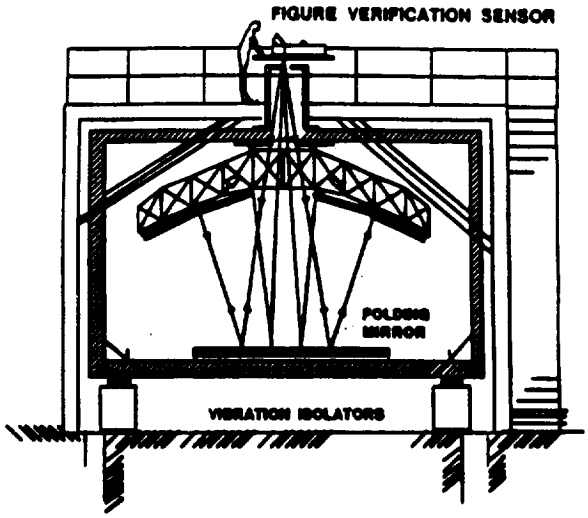


Fig.4 PSR TD Configuration

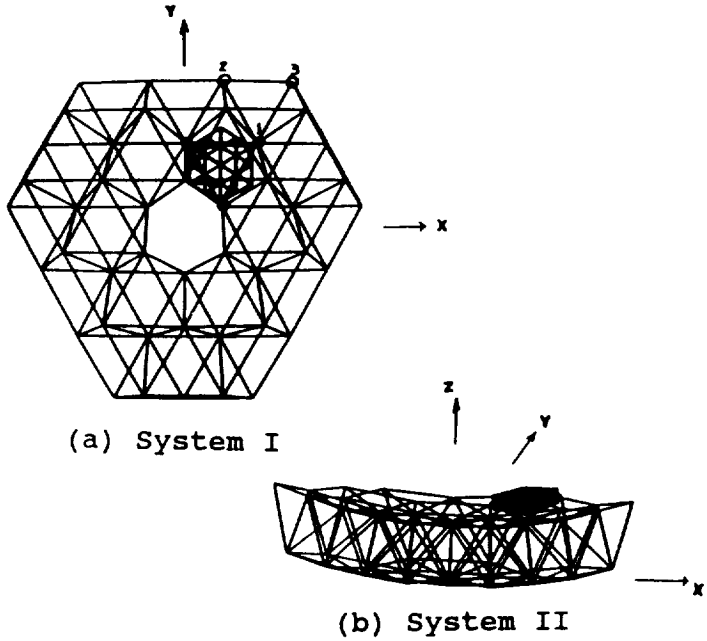


Fig.5 PSR TD System Models

The natural frequencies of these two system models are listed in Table 1, with the corresponding mode shapes briefly described. It should be noted that the natural frequencies of the APM alone are very close to those of the PSR system models. No couplings are observed for the piston mode, tilt modes and rotational mode between the APM and the backup struts. However, slight couplings are noted for the lateral modes. This is may also be due to the effects of an in-plane offset as discussed in Ref. 5.

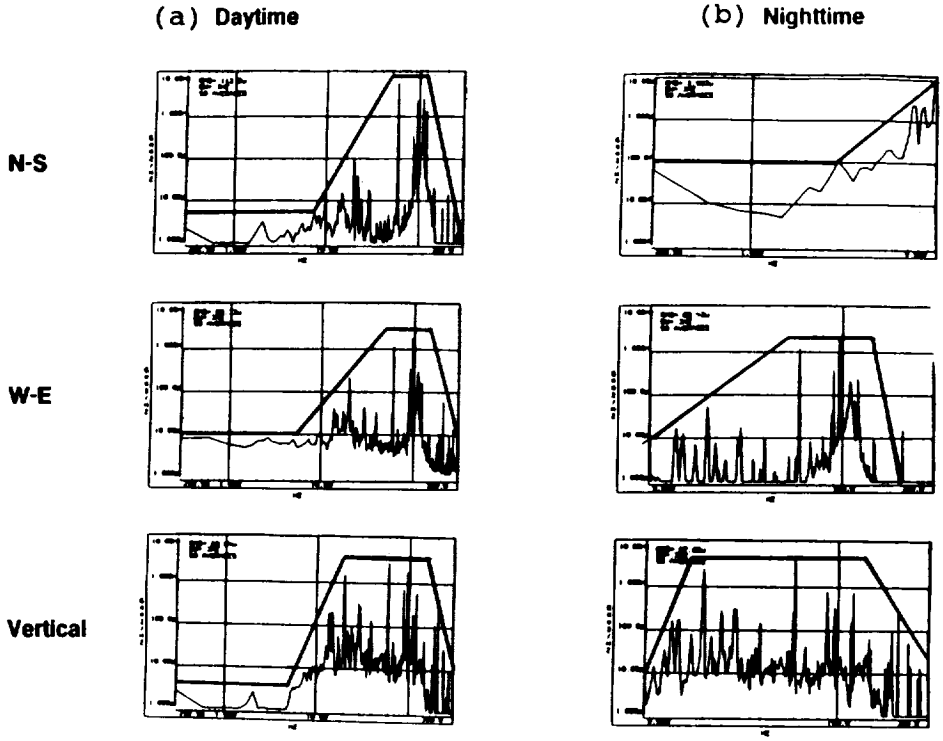
Table 1 Dynamic Characteristics of PSR System Models

MODE NO.	NATURAL FREQUENCY (Hz)		MODE SHAPE
	SYSTEM I	SYSTEM II	
1-6	0.000	0.000	Rigid Body Modes
7	0.087	0.087	Panel Piston Mode
8	0.105	0.105	Panel Tilt Mode
9	0.106	0.106	Panel Tilt Mode
10	25.15	25.32	Bending Mode of Backup Struts
11	25.29	25.75	Bending Mode of Backup Struts
12	29.43	29.43	Panel Core Mode
13	43.53	44.31	Translation Mode in X-direction
14	53.05	47.32	Translation Mode in Y-direction
15	54.12	53.06	
16	55.62	55.47	Panel Rotational Mode
17	57.46	57.55	
18	67.44	67.09	
19	68.35	68.90	
20	95.79	96.84	

CHARACTERIZATION OF THE TEST ENVIRONMENT

The PSR TB structural tests are to be performed in the Magnet Room of High Bay 1 located in the Spacecraft Assembly Facility (SAF) at JPL. A survey was conducted to characterize acoustic and seismic environments of this proposed test site [7]. In this survey, acoustic and seismic data were accumulated over a time period of one week. For ground motion measurements, three Wilcox Research Model 731 accelerometers, one unit along each of the north-south, east-west, and vertical axes, were used. The 1dB frequency responses of these seismic accelerometers were measured from 0.1 to 300 Hz. Three set of data, for day time, night time, and day time with equipment off, were collected by these accelerometers. The collected data was presented in three forms: (1) G^2/Hz vs. Hz; (2) G vs. Hz; and (3) peak displacement vs. Hz. A 1024 point Fast Fourier Transform was taken with a 1024-channel analyzer to convert the raw data into frequency domain from the time domain. The resulted acceleration power spectrum densities of the measured seismic disturbances are applied in the random response analyses of the PSR Technology Demonstration system model. Two extreme cases are examined in this work: (1) daytime disturbances (Fig. 6.a), and (2) nighttime disturbances (Fig. 6.b). The coordinate system shown in Figure 6 is defined as follows: X-Axis is for the recorded north-south data, Y-Axis is for the east-west direction and Z-Axis is for the vertical direction. For conservative purposes, the envelopes shown in these disturbances are applied in the random analyses. It is noted that the magnitudes of the daytime disturbances in the low frequency range are much higher than those of the nighttime disturbances. However, the magnitudes of the daytime disturbance in the high frequency range are very close to those of the nighttime disturbances.

Fig.6 **SAF DISTURBANCES**



ANALYTICAL APPROACH

Random analysis approach used in this work is based on a data reduction procedure that is applied to the results of a frequency response analysis. The frequency response function H(f) is obtained by applying a variable frequency sinusoidal acceleration, A_o, to the PSR system models and calculating the acceleration response at the specified points. Dividing the calculated acceleration by the input A_o, H(f) can be expressed as function of the excitation frequency, f. Then the root-mean-square (RMS) responses ($\bar{a}$) at the specified points can be calculated numerically from the equation

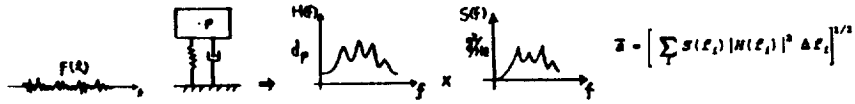
$$\bar{a} = \left[\sum_i S(f_i) |H(f_i)|^2 \Delta f_i \right]^{1/2}$$

where $S(f_i)$ is the acceleration power spectral density function at the discrete frequency f_i.

The random response analyses are implemented by using the NASTRAN modal frequency response solution scheme (Sol. 30) coupled with the results from the normal mode analyses (Sol. 3). The peak random responses of the PSR system are calculated by using the RMS values of frequency responses induces by random disturbances over a frequency range form 0.01 Hz to 200 Hz. An uncorrelated approach is applied in this work in order to be able to examine the peak responses of the PSR system due to each individual external disturbance in a different excitation axis. The final peak responses of the system subjected to the disturbances of all three axes are then calculated by using the root sum square (RSS) of the RMS peak responses in three axes. Two sets of relative displacements are calculated in the analyses. The first one is the relative displacement between the grid point of the front panel facesheet, located above the truss node, and the backup truss node. The second set is the relative displacement between the backup truss node and the ground support points. An 0.5% modal damping was applied to the frequency response analyses.

RANDOM RESPONSE ANALYSES

- **Modal frequency response associated with results from normal mode analyses**
- **Data reduction procedure**



- **Implemented by using MSC/NASTRAN**
- **Uncorrelated approach**
- **Frequency range: 0.01Hz to 200 Hz**
  Modal damping: 0.5%
- **Probability of exceeding the specific displacement**

RESULTS AND DISCUSSION

The results of the 1-σ peak displacements are summarized is Table 2 for both the PSR System I and System II models subjected to these seismic disturbances. It is noted that the movements of the panel occurred in nighttime are much smaller than those occurred in daytime. However, the difference in the nighttime and the daytime movements of the nodes is not as large as that in the panel. This is because the movements of the panel are predominant in the lower frequency (about 0.1 Hz) range (Figure 7) and the movements of the nodes are predominant at a higher frequency (about 25 Hz) governed by the truss modes (Figure 8). It is also noted that the lateral movements of the panel are larger in the PSR System II than those in the PSR System I. However, the vertical movements of the panel are almost identical in both PSR systems. This is because the vertical movements of the panel are dominated by the piston modes and the natural frequencies of the piston mode in both PSR systems are identical. Another observation is that the lateral movements of the panel are more location-dependent than the vertical movements of the panel. However, the opposite results are observed in the movement of the strut nodes.

For the proposed PSR Technology Demonstration configuration (System I), the lateral peak movements (1-σ) are about 2.9 μm for the daytime disturbance case and 0.36 μm for the nighttime case. The vertical peak movements are 13μm and 2μm for the daytime case and the nighttime case, respectively. The 1-σ peak responses of the PSR System II are 4.5 μm for the lateral movement and 13 μm for the vertical movement in the daytime case. These peak responses are well below the expected figure control range of $\pm$1mm and the 100 μm gaps between the panels. This implies that the isolation table is not a necessity in the TD optical bench design.

The peak responses of a hard mount case had also been studied by increasing the flexure size of the lateral constraint struts such that the natural frequency of the piston mode is 1.6 Hz and the natural frequency of the tilt modes is 2.0 Hz. Results of analyses based on the nighttime SAF environmental data indicate that the 1-σ lateral movement of the panel is 0.14 μm and the 1-σ vertical movement of the panel is 0.08 μm. The comparison indicates that the peak vertical movement can be reduced dramatically (from 2.0 μm to 0.08 μm) by using the hard mount APM design.

Table 2 1-σ RMS Displacement Responses (μ m) of the PSR systems

Location	Component	Daytime Disturbances		Nighttime Disturbances	
		System 1	System 2	System 1	System 2
Relative Displacements Between Front Panel Facesheet and Strut Nodes					
Corner 1	x	1.34	2.18	0.202	0.377
	y	1.72	3.14	0.293	0.479
	z	12.89	12.85	1.851	1.794
Corner 2	x	1.12	1.89	0.172	0.284
	y	2.58	4.05	0.336	0.573
	z	12.53	12.46	1.911	1.846
Corner 3	x	1.50	2.29	0.221	0.336
	y	2.54	3.92	0.331	0.551
	z	11.96	11.83	1.819	1.751
Relative Displacements Between Strut Nodes and Ground					
Node 1	x	0.026	0.031	0.019	0.022
	y	0.018	0.019	0.013	0.015
	z	0.012	0.038	0.010	0.032
Node 2	x	0.029	0.034	0.020	0.024
	y	0.018	0.019	0.013	0.015
	z	0.022	0.044	0.020	0.039
Node 3	x	0.032	0.037	0.023	0.027
	y	0.023	0.027	0.017	0.022
	z	0.053	0.065	0.046	0.055

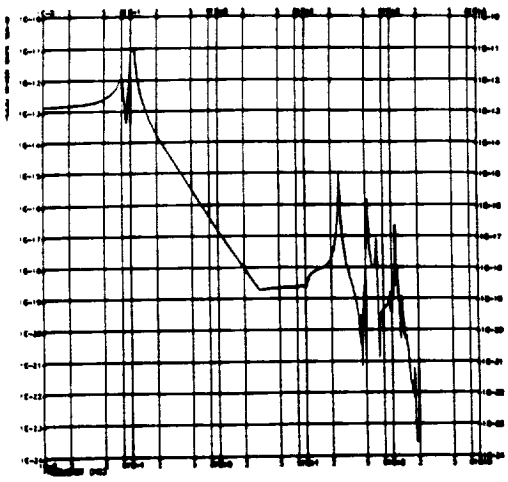


Fig.7 Response of Panel Facesheet

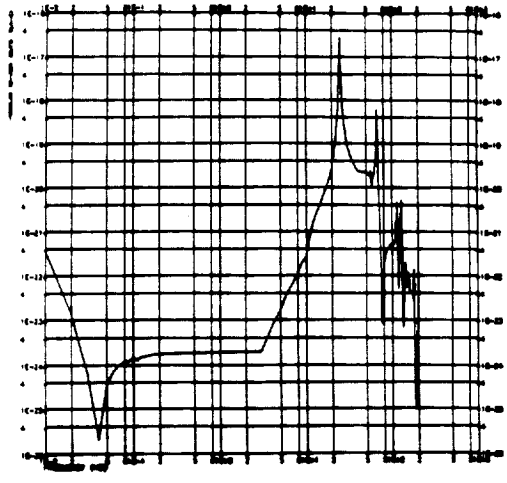


Fig.8 Response of Strut Node

CONCLUSION

Technologies, including those related to large space structures, developed by the PSR program play a vital role in enabling future astronomical missions that require large precise telescopes. To verify these enabling technologies, ground tests must be performed and the planning of the tests mandates a need for a thorough assessment of the test environment and responses of the test structure to the environment. This need has been partially satisfied by random vibration analyses of the PSR structure using seismic inputs derived from measurements of ground motions of the test site. Results of the analyses indicated that the maximum daytime movements of the precise panel supported by the PSR structural system, including the APM, will be less than 13 microns in the vertical direction and 3 microns in the lateral directions. These movements are well within the acceptable limits and the need for elaborate vibration isolation devices does not exist. The next step in planning the PSR TD tests is to design an optical-bench structure which will not amplify or adversely alter the seismic disturbances imposed on the test structure. The PSR TD optical bench will be extremely stiff such that frequencies of its vibratory modes are well above the frequency range occupied by PSR structural system. A fundamental frequency above 50 Hz is considered to be desirable for the PSR TD optical bench. Design of such an optical bench is currently in progress.

SUMMARY

- **PSR Technology Demonstration system model has been established**
  - **Panel**
  - **Articulated Panel Module (APM)**
  - **Backup truss**
- **Seismic disturbances of the PSR TD test site were measured. The resulted acceleration power spectrum densities of these disturbances were applied in the random response analyses of the PSR TD system model.**
- **Analytical results indicated that the movements of the precise panels supported by the PSR structural system were within the acceptable limits.**
- **Elaborate vibration isolation devices are not necessary.**
- **Future Work**
  - **Optical bench**
  - **Suspension system**

REFERENCE

1. Swanson, P. N., "A Lightweight Low Cost Large Deployable Reflector (LDR)," JPL Report D-2283, June, 1985.

2. Collins, T. J. and Fichter, W. B., "Design of Support Trusses for Large PSRs," presented at SPIE Conference on Active Telescope Systems, Orlando, FL, March, 1989.

3. Bush, H., et al., "Design and Fabrication of an Erectable Truss for Precision Segmented Reflector Application," presented at the 31st AIAA/ASME/ASCE/AHS/ ACS Structures, Structural Dynamics and Materials Conference, Long Beach, CA, April, 1990, Paper No. 90-0999.

4. JPL IOM 343-89-318 Rev. A.,E. Mattler to M. Lou/C. F. Shih, Subject: Interface Requirements Update for the Prototype APM, dated August 31,1989.

5. Shih, C. F., Moore, D., Olivera, A., and Lou, M.,"Design and Analysis of Precision Segmented Reflector (PSR) Panel Attachment," JPL Report D-7069, January, 1990.

6. Kuo, C. P. and Wada, B. K., "Composite Deformable Mirror," presented at SPIE Conference on Active Telescope Systems, Orlando, FL, March, 1989.

7. Cohen, E. J. and Parks, G. S.,"SAF Seismic and Acoustical Data," JPL Interoffice Memorandum, dated May 15, 1989.

Flexible Body Dynamic Stability for High Performance Aircraft

N91-22318

By

E. A. Goforth*, H. M. Youssef**, C. V. Apelian†, S. C. Schroeder††
Lockheed Aeronautical Systems Company
P.O.Box 551
Burbank, CA 91520-7550

ABSTRACT

Dynamic equations which include the effects of unsteady aerodynamic forces and a flexible body structure have been developed for a free-flying high performance fighter aircraft. The linear and angular deformations are assumed to be small in the body reference frame, allowing the equations to be linearized in the deformation variables. Equations for total body dynamics and flexible body dynamics are formulated using the hybrid coordinate method and integrated in a state space format. A detailed finite element model of a generic high-performance fighter aircraft is used to generate the mass and stiffness matrices. Unsteady aerodynamics are represented by a rational function approximation of the doublet lattice matrices. The equations simplify for the case of constant angular rate of the body reference frame, allowing the effect of roll rate to be studied by computing the eigenvalues of the system. It is found that the rigid body modes of the aircraft are greatly affected by introducing a constant roll rate, while the effect on the flexible modes is minimal for this configuration.

1. INTRODUCTION

Future fighter aircraft must be able to meet stringent maneuverability and performance requirements. This will result in aircraft designs in which the interaction of flexibility, aerodynamics, and overall body motion during a maneuver are of prime importance. The need for superagility and the use of advanced lightweight materials will make it very important to consider flexibility effects in the analysis of the aircraft undergoing maneuvers at high rates.

Flexible body dynamics have been investigated in many other writings, including references [1] - [4]. In this paper, dynamic equations will be derived in a manner similar to that in reference [5], which contains a more thorough development of the equations. In addition, aerodynamic forces will be explicitly included in the equations. These equations will then be applied to a realistic model of a modern fighter aircraft.

The aircraft is assumed to be a collection of elastically interconnected, discrete rigid subbodies which are subjected to external forces and torques, including unsteady aerodynamic forces. It is assumed that the deformations of the subbodies with respect to the body reference frame are small so that the high order terms in the deformation variables and their rates can be neglected. The rotational effects of motors, fans, and turbines are not included in this representation.

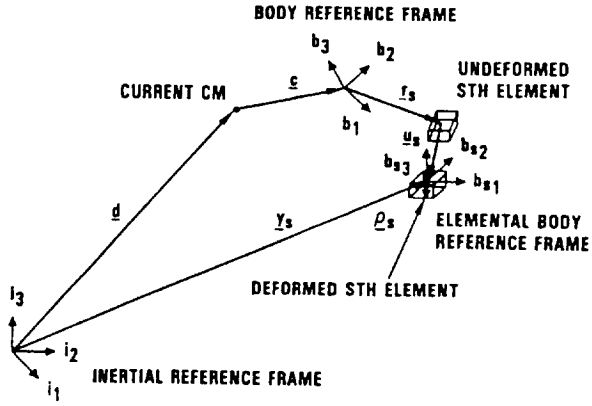


Figure 1. Reference Frames and a Subbody

*Senior Research Specialist
**Department Engineer
†Associate Engineer
††Senior Associate Engineer

Table 1. Vectors and Dyads definitions

Vectors and Dyads	Definition
$\{i\}^T$	basis matrix for inertial reference frame (IRF)
$\{b\}^T$	basis matrix for body reference frame (BRF)
$\{i\}^T = \{b\}^T \theta$	direction cosine matrix relationship between IRF and BRF
$\omega = \{b\}^T \omega$	angular velocity of body reference frame
$\{b_s\}^T$	basis matrix for sth elemental body reference frame(ERF)
$d = \{i\}^T d$	Position of current center of mass (CM) in IRF
$\bar{V} = \{i\}^T V$	Velocity of current CM in IRF
$\underline{Y}_s = \{i\}^T Y_s$	Position of the sth element in IRF
$\underline{c} = \{b\}^T c$	position of CM in BRF
$\underline{r}_s = \{b\}^T r_s$	position of undeformed sth element from undeformed CM
$\underline{u}_s = \{b\}^T u_s$	position of sth element from the undeformed position
$\underline{\rho}_s = \{b_s\}^T \rho_s$	position of differential mass in sth elemental body
$\underline{\beta}_s = \{b\}^T \beta_s$	angular deformation of sth element
$\underline{\underline{I}} = \{b\}^T I \{b\}$	Inertia dyadic of the aircraft with respect to CM
$\underline{\underline{I}}_s = \{b_s\}^T I_s \{b_s\}$	Inertia dyadic of the sth elemental body about its CM

Table 2. Vector Identities and Matrix Operation Equivalents

Vector representation:

$$\underline{x} = \{i\}^T \begin{bmatrix} x1 \\ x_2 \\ x_3 \end{bmatrix} = \{i\}^T X$$

$$\underline{y} = \{i\}^T \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \{i\}^T Y$$

Cross product representation:

$$\underline{x} \times \underline{y} = \{i\}^T \tilde{X} Y = - \{i\}^T \tilde{Y} X$$

where the $\tilde{}$ operator is defined below:

$$(X)^{\tilde{}} = \tilde{X} = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}$$

Operations with dyad

$$\underline{\underline{I}} = \{i\}^T I \{i\}$$

$$\underline{\underline{I}} \cdot \underline{x} = \{i\}^T I X$$

$$\underline{\underline{I}} \times \underline{x} \cdot \underline{y} = \{i\}^T I \tilde{X} Y$$

A highly detailed description of the hybrid coordinate method, which is used here to develop the dynamic equations, can be found in references [5] and [6]. Only some highlights of the development of these equations will be presented in this paper. The development of the equations closely follows that of reference [7], with aerodynamic forces added. The equations are implemented as a computer program, FLXAIR.

Figure 1 shows a schematic diagram of the various reference frames associated with each subbody. Definitions of the vectors and dyads used in this figure and in the derivations are given in Table 1. Table 2 shows the equivalency between various operations in a vector/dyad format and those in a matrix format. The matrix format is used for implementing the computer solution to our problem.

Section 2 deals with the net force and torque applied to the total body. The net forces and torques on the subbodies are described in section 3. Derivations are kept brief, with only main steps provided. The rational function approximation used for describing the unsteady aerodynamic forces is given in section 4. In section 5 the equations are integrated in a state space format, with aerodynamic forces specifically separated from other external forces. These nonlinear and time dependent equations can be used for simulation. When the angular velocity of the body reference frame is constant, the equations become time invariant. It is then possible to study the effects of angular velocity on vehicle structural dynamics by performing an eigenvalue analysis. When unsteady aerodynamic loading is included in this formulation, this is seen to be a flutter analysis under maneuver.

A large-order finite element model which is a realistic representation of an advanced fighter was used to demonstrate the stability effects of high roll rates. Section 6 describes the NASTRAN

model, structural and aerodynamic, which was used in the analysis. Results are shown for various roll rates and for variations of overall stiffness of the aircraft. The analyses show little effect on the flexible modes of the system due to roll maneuvers. Considerable effect was, however, observed for the rigid body modes.

2. TOTAL BODY DYNAMICS

The equations are derived from Newton-Euler equations. The equations for the net force $\underline{F}$, and the net torque $\underline{P}$, can be represented as follows.

$$\underline{F} = {}^{i}\frac{d^2}{dt^2}(M_{Tot}\ \underline{d}) \tag{1}$$

$$\underline{P} = {}^{i}\frac{d}{dt}(\underline{H}) \tag{2}$$

M_{Tot} is the total mass, and $\underline{H}$ is the angular momentum referred to the CM of the aircraft. Presuperscript i refers to the fact that the differentiation must be with respect to the inertial reference frame.

The further development neglects the effects of rotating bodies such as engine compressors, fans, rotors, etc.. It is assumed that the deformation of flexible bodies is small in the body reference frame. This assumption is used to neglect the high order terms in the deformation variables $\underline{u}_s$(linear deformation of sth element) and $\underline{\beta}_s$(angular deformation of the sth element) and their derivatives.

Equation (1) can be written in the body reference frame and for ease of computer implementation in matrix form as follows:

$$F = M_{Tot}\ \theta\dot{V} \tag{3}$$

The development of equation (2) to a computer implementable step is lengthy. Only few key steps are given.
The angular momentum $\underline{H}$ is defined as

$$\underline{H} = \int(\underline{c}+\underline{r}_s+\underline{u}_s+\underline{\rho}_s) \times {}^{i}\frac{d}{dt}(\underline{c}+\underline{r}_s+\underline{u}_s+\underline{\rho}_s)\ dm \tag{4}$$

The development makes use of the mass-center definition

$$\int(\underline{c}+\underline{r}_s+\underline{u}_s+\underline{\rho}_s)\ dm = 0 \tag{5}$$

and the following identity:

$$\int(\underline{r}_s+\underline{u}_s+\underline{\rho}_s) \times (\ \underline{\omega} \times (\underline{r}_s+\underline{u}_s+\underline{\rho}_s)\ )\ dm = \underline{I}\cdot\underline{\omega} \tag{6}$$

With the use of equations (4) to (6), equation (2) can be written as

$$\underline{P} = \underline{I}\cdot\dot{\underline{\omega}} + \underline{\omega} \times \underline{I}\cdot\underline{\omega} + \dot{\underline{I}}\cdot\underline{\omega} + \ddot{\underline{c}} \times \underline{c}$$

$$+ {}^{i}\frac{d}{dt}\int(\underline{r}_s+\underline{u}_s+\underline{\rho}_s) \times (\dot{\underline{r}}_s+\dot{\underline{u}}_s+\dot{\underline{\rho}}_s)\ dm \tag{7}$$

The assumptions of discrete lumped masses and small deformations with respect to the body reference frame are now used to convert the integration operation into the following summation operation:

$$\int(\underline{r}_s+\underline{u}_s+\underline{\rho}_s) \times (\dot{\underline{r}}_s+\dot{\underline{u}}_s+\dot{\underline{\rho}}_s)\ dm =$$

$$\Sigma\underline{r}_s \times m_s\ \dot{\underline{u}}_s + \Sigma\ \underline{I}_s\cdot\dot{\underline{\beta}}_s \tag{8}$$

where m_s is defined as:

$$\int_s dm = m_s \tag{9}$$

and center of mass definition of sth lumped mass is given by

$$\int_s \underline{\rho}_s\ dm = 0 \tag{10}$$

Finally, equation (7) can be written in a computer implementable form as:

$$P = I\dot{\omega} + \dot{I}\omega + \tilde{\omega}I\omega$$

$$+ \Sigma\{((\tilde{\omega}r_s)^{\sim} + \tilde{r}_s\tilde{\omega})\dot{u}_s m_s + \tilde{r}_s\ddot{u}_s m_s + I_s\ddot{\beta}_s\}$$

$$+ \tilde{\omega}\Sigma I_s\dot{\beta}_s \tag{11}$$

The total inertia is assumed to be linear in the deformation variables.

$$I = I^* + \Sigma m_s\{2r_s^T u_s E - r_s u_s^T - u_s r_s^T\}$$

$$+ \Sigma(\tilde{\beta}_s I_s - I_s\tilde{\beta}_s) \tag{12}$$

where I^* is the inertia of the undeformed airplane. Therefore

$$\dot{I} = \Sigma m_s\{2r_s^T\dot{u}_s E - r_s\dot{u}_s^T - u_s\dot{r}_s^T\}$$

$$+ \Sigma(\dot{\tilde{\beta}}_s I_s - I_s\dot{\tilde{\beta}}_s) \tag{13}$$

Equations (12) and (13) can be substituted into equation (11) to further simplify the equation.

3. ELEMENTAL BODY DYNAMICS

The net forces and torques on the sth elemental body are as follows:

$$\underline{f}_s = m_s \, {}^{i}\frac{d^2}{dt^2}(\underline{d} + \underline{c} + \underline{r}_s + \underline{u}_s) \tag{14}$$

$$\underline{p}_s = {}^{i}\frac{d}{dt}\underline{H}_s \tag{15}$$

$\underline{H}_s$ is the inertial angular momentum of sth element referred to its mass center. Note that this equation is applicable to all n subbodies.

It is assumed that the body reference frame and the elemental body reference frames are initially colinear. This assumption, though not necessary, is used here to simplify the equations.

Equation (14) can be written in the following computer implementable form:

$$f_s = m_s \{ \theta\dot{V} + \ddot{c} + \dot{\tilde{\omega}}(r_s+u_s+c) + 2\tilde{\omega}(\dot{c}+\dot{u}_s) + \ddot{u}_s + \tilde{\omega}\tilde{\omega}(r_s+u_s+c) \} \tag{16}$$

noting that

$$V = \frac{d}{dt}(d) \tag{17}$$

$\underline{H}_s$ is defined as

$$\underline{H}_s = \underline{I}_s \cdot \underline{\omega} \tag{18}$$

Invoking the assumption of small deformation, the rotation is represented by

$$\underline{\beta}_s = [b]^T \beta_s \tag{19}$$

Note that this equation is strictly true if the rotations are infinitesimally small. The relationship between the body reference frame and the elemental reference frame can now be approximated as

$$[b_s]^T = [b]^T (E+\tilde{\beta}_s) \tag{20}$$

where E is a 3x3 unit matrix.

Using equations (18) to (20), equation (15) can be written in the following computer implementable form:

$$p_s = I_s(\dot{\omega}+\ddot{\beta}_s) + (I_s\tilde{\omega}+\tilde{\omega}I_s-(I_s\omega)^{\sim})\dot{\beta}_s + \tilde{\omega}I_s\omega + (I_s\dot{\tilde{\omega}} - (I_s\dot{\omega})^{\sim} - \tilde{\omega}(I_s\omega)^{\sim} + \tilde{\omega}I_s\tilde{\omega})\beta_s \tag{21}$$

4. RATIONAL FUNCTION APPROXIMATION OF UNSTEADY AERODYNAMICS

The formulation of the unsteady aerodynamics is based on the relation

$$\{\frac{w}{V}\} = \frac{2}{\rho V^2} [NID] \{\Delta p\} \tag{22}$$

where $\{\Delta p\}$ represents the pressures at aerodynamic force nodes, $\{w\}$ contains the velocities normal to the lifting surface induced by $\{\Delta p\}$, and [NID] is the induced normal downwash influence matrix. The induced velocities are defined as downwash collocation points which are located at the 3/4 chord of each aerodynamic box for the doublet lattice method.

Downwash collocation points are those points on a lifting surface at which the induced velocity normalized by the free stream velocity is equal to the local angle of attack $\{a\}$, i.e.,

$$\{a\} = \{\frac{w}{V}\} \tag{23}$$

The pressures are then given by

$$\{\Delta p\} = \frac{1}{2} \rho V^2 [NID]^{-1} \{a\} \tag{24}$$

or

$$\{\Delta p\} = \frac{1}{2} \rho V^2 [AIC] \{a\} \tag{25}$$

where $[AIC] = [NID]^{-1}$.

In the following derivation, Equation (24) is used as a starting point.

From $\{\Delta p\}$, by an integration or "lumping" process represented by [ZP], the aerodynamic forces are obtained:

$$\{Z_{aero}\} = [ZP] \{\Delta p\} \tag{26}$$

The local angle of attack, taken relative to the free stream velocity, V, is given by

$$\{a\} = \{a_T\} + \{a_z\} \tag{27}$$

The contribution a_T is the instantaneous slope of the lifting surface, relative to V, in a plane through V perpendicular to the lifting

surface:

$$\{a_T\} = [D_\theta]\{z\} \tag{28a}$$

where $[D_\theta]$ is a differentiating matrix.

The contribution $a_{\dot{z}}$ results from the rate of translation in a direction perpendicular to the lifting surface:

$$\{a_{\dot{z}}\} = [D_z]\left\{\frac{\dot{z}}{V}\right\} \tag{28b}$$

where $[D_z]$ is an interpolating matrix.

Substituting Equation (28) into Equation (27) and replacing $\dot{z}$ by sz yields:

$$\{a\} = \left[[D_\theta] + \frac{s}{V}[D_z]\right]\{z\} \tag{29}$$

Combining Equations (24), (26) and (29) leads to:

$$\{Z_{aero}\} = \frac{1}{2}\rho V^2[ZP][NID]^{-1} * \left[[D_\theta] + \frac{s}{V}[D_z]\right]\{z\} \tag{30}$$

For constant amplitude oscillation $s = i\omega = i(Vk/c)$. The induced velocity matrix is a function of ik. It follows that Equation (30) can be written as:

$$\{Z_{aero}\} = \frac{1}{2}\rho V^2[A(ik)]\{z\} \tag{31}$$

where A(ik) is given by:

$$[A(ik)] = [ZP][NID(ik)]^{-1}\left[[D_\theta] + \frac{ik}{c}[D_z]\right] \tag{32}$$

For developing the explicit function of s, $[A(s)]$, corresponding to $[A(ik)]$, the $[D_\theta]$ and $[D_z]$ contribution to $[A(ik)]$ are identified separately, and the explicit occurrence of s in Equation (30) is maintained.

$$[A(ik,s)] = [ZP][NID(ik)]^{-1} * [D_\theta] + \frac{s}{V}[ZP][NID(ik)]^{-1}[D_z] \tag{33}$$

Let:

$$[A_T(ik)] = [ZP][NID(ik)]^{-1}[D_\theta] \tag{34}$$

and

$$[A_z(ik)] = [Zp][NID(ik)]^{-1}[D_z] \tag{35}$$

Then:

$$[A(ik,s)] = [A_T(ik)] + \frac{s}{V}[A_z(ik)] \tag{36}$$

Preliminary to approximating $[A(ik,s)]$ by an explicit function of only s, $[A_T(ik)]$ and $[A_z(ik)]$ are approximated by $[A_T(p)]$ and $[A_z(p)]$, where p is the nondimensional form of s: $p = cs/V$.

Following Reference [8], the following terms are approximately,

$$[A_T(p)] = [B_{T0}] + \sum_{j=1}^{n}\frac{[B_{Tj}]p}{p + b_j} \tag{37}$$

$$[A_z(p)] = [B_{z0}] + \sum_{j=1}^{n}\frac{[B_{zi}]p}{p + b_j} \tag{38}$$

These matrices can be obtained by generating aerodynamic matrices for several values of k and then employing a least-squares fit.

Because the state-space equation will be written in terms of s, Equations (37) and (38) are written in terms of s by letting $p = cs/V$:

$$[A_T(s)] = [B_{T0}] + s\sum_{j=1}^{n}\frac{[B_{Tj}]}{s + \beta_j} \tag{39}$$

$$[A_z(s)] = [B_{z0}] + s\sum_{j=1}^{n}\frac{[B_{zj}]}{s + \beta_j} \tag{40}$$

where

$$\beta_j = V b_j/c$$

Combination of Equations (31), (35), (36), (39), and (40) leads to the following approximate expressions for the aerodynamic forces:

$$\{Z_{aero}\} = \frac{1}{2}\rho V^2\left[[B_{T0}] + s\sum_{j=1}^{n}\frac{[B_{Tj}]}{s + \beta_j}\right]\{z\} + \frac{1}{2}\rho V^2\left[\frac{s}{V}[B_{z0}] + \frac{s^2}{V}\sum_{j=1}^{n}\frac{[B_{zj}]}{s + \beta_j}\right]\{z\} \tag{41}$$

5. INTEGRATED TOTAL BODY AND ELEMENTAL BODY DYNAMICS

The general form of the linear flexible body equation is

$$M'\ddot{q} + D\dot{q} + G'(\tilde{\omega})\dot{q} + K'(\tilde{\omega}\tilde{\omega},K)q + A'(\dot{\tilde{\omega}},\tilde{\omega})q = L'(\dot{\tilde{\omega}},\tilde{\omega},f_s,p_s) \quad (42)$$

where $q=[u_1^T,\beta_1^T,..,u_s^T,\beta_s^T,..,u_n^T,\beta_n^T]^T$. M, D, and K are mass, damping, and stiffness matrices respectively of the airplane which are obtained by a traditional finite element method, such as NASTRAN. The other terms, G', K', A' and L' are obtained from equations (16), and (42). Note that K' is a symmetric matrix. Equations (3), (11), (12), (13) and (42) can be written in the state space like format as follows:

$$\begin{bmatrix} M_{Tot}\theta & 0 & 0 & 0 \\ 0 & I^* & 0 & \Gamma_0 \\ 0 & 0 & E_n & 0 \\ M\Sigma_{E0}\theta & \Lambda_0 & 0 & M' \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\omega} \\ \dot{\eta}_1 \\ \dot{\eta}_2 \end{bmatrix} +$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \tilde{\omega}I^* & \bar{\Sigma} & \Gamma_1 \\ 0 & 0 & 0 & -E_n \\ 0 & \Lambda_1 & K'+A' & G'+D \end{bmatrix} \begin{bmatrix} V \\ \omega \\ \eta_1 \\ \eta_2 \end{bmatrix} = \begin{bmatrix} F \\ P \\ 0 \\ \lambda \end{bmatrix} \quad (43)$$

Equations (44)-(55) explain various terms in equation (43).

$$\eta_1 = q \quad (44)$$

$$\eta_2 = \dot{q} \quad (45)$$

0 is a null matrix and dimensions are context dependent.

E_n is 6nx6n matrix.

$$\Gamma_0\dot{\eta}_2 = \Sigma m_s\tilde{r}_s\ddot{u}_s + \Sigma I_s\ddot{\beta}_s \quad (46)$$

$$\bar{\Sigma}\eta_1 = \tilde{\omega}\{\Sigma m_s(2r_s^Tu_sE - r_su_s^T - u_sr_s^T) + \Sigma\tilde{\beta}_sI_s - \Sigma I_s\tilde{\beta}_s\}\omega + \{\Sigma m_s(2r_s^Tu_sE - r_su_s^T - u_sr_s^T) + \Sigma\dot{\tilde{\beta}}_sI_s - \Sigma I_s\dot{\tilde{\beta}}_s\}\omega \quad (47)$$

$$\Gamma_1\eta_2 = (\Sigma m_s(2r_s^T\dot{u}_sE - r_s\dot{u}_s^T - \dot{u}_sr_s^T) + \Sigma\dot{\tilde{\beta}}_sI_s - \Sigma I_s\dot{\tilde{\beta}}_s)\omega + \Sigma m_s[(\tilde{\omega}r_s)^- + \tilde{r}_s\tilde{\omega}]u_s + \Sigma\tilde{\omega}I_s\beta_s \quad (48)$$

E = 3x3 unit matrix

M = block diagonal 6nx6n matrix where block diagonals are 3x3 matrices

= Block diagonal [m_1E, I_1, .., m_sE, I_s, .., m_nE, I_n]

Σ_{E0} = [E 0 E 0 .. E 0]^T 6nx3 matrix

Σ_{0E} = [0 E 0 E .. 0 E]^T 6nx3 matrix

$$M' = M(E_n - \Sigma_{E0}\Sigma^T_{E0}M/M_{Tot}) \quad (49)$$

$$\Lambda_0 = M(\Sigma_{0E} - \tilde{R}) \quad (50)$$

$$\tilde{R} = \begin{bmatrix} \tilde{r}_1 \\ 0 \\ \tilde{r}_2 \\ 0 \\ \vdots \\ \tilde{r}_n \\ 0 \end{bmatrix} \quad (51)$$

$$\Lambda_1 = [\Sigma_{0E}\omega]^-M\Sigma_{0E} - M[\Sigma_{E0}\omega]^-\tilde{R} \quad (52)$$

$[\Sigma_{0E}\omega]^-$ = block diagonal matrix of dimension 6nx6n. Each block is 3x3.
The diagonal blocks are
[$\tilde{\omega}$ 0 $\tilde{\omega}$ 0 .. $\tilde{\omega}$ 0]

$[\Sigma_{E0}\omega]^-$ = block diagonal matrix of dimension 6nx6n. Each block is 3x3.
The diagonal blocks are
[0 $\tilde{\omega}$ 0 $\tilde{\omega}$ .. 0 $\tilde{\omega}$]

$[M\Sigma_{0E}\omega]^-$ = block diagonal matrix of dimension 6nx6n. Each block is 3x3.
The diagonal blocks are
[0 $(I_1\omega)^-$ 0 $(I_2\omega)^-$ .. 0 $(I_n\omega)^-$]

$$K' = K + [\Sigma_{0E}\omega]^-M[\Sigma_{0E}\omega]^- + M\{[\Sigma_{E0}\omega]^- * [\Sigma_{E0}\omega]^- - \Sigma_{E0}\tilde{\omega}\tilde{\omega}\Sigma^T_{E0}M/M_{Tot}\} \quad (53)$$

$$A' = M[\Sigma_{OE}\dot{\omega}]^{\sim} - [M\Sigma_{OE}\dot{\omega}]^{\sim} + M\{[\Sigma_{EO}\dot{\omega}]^{\sim} - \Sigma_{EO}\dot{\tilde{\omega}}\Sigma_{EO}M/M_{Tot}\} + [\Sigma_{OE}\omega]^{\sim}[M\Sigma_{OE}\omega]^{\sim} \quad (54)$$

$$G' = M[\Sigma_{OE}\omega]^{\sim} + [\Sigma_{OE}\omega]^{\sim}M - [M\Sigma_{OE}\omega]^{\sim} + 2M\{[\Sigma_{EO}\omega]^{\sim} - \Sigma_{EO}\tilde{\omega}\Sigma_{EO}M/M_{Tot}\} \quad (55)$$

Definitions of $[\Sigma_{OE}\dot{\omega}]^{\sim}$, $[\Sigma_{EO}\dot{\omega}]^{\sim}$, and $[M\Sigma_{OE}\dot{\omega}]^{\sim}$ are very similar to $[\Sigma_{OE}\omega]^{\sim}$, $[\Sigma_{EO}\omega]^{\sim}$, and $[M\Sigma_{OE}\omega]^{\sim}$ and hence they are not given here.

Equation (43) has the form

$$A_0\,\dot{X} + A_1\,X = U \quad (56)$$

where definitions of A_0, A_1, X, and U are obvious. Equation (56) can be written as

$$\dot{X} = -A_0^{-1}\,A_1\,X + A_0^{-1}\,U \quad (57)$$

Eq.(57) can be simply written as

$$\dot{X} = A\,X + B\,U \quad (58)$$

Definitions of A and B are obvious.

Eq.(58) can be used for the time simulation. To better understand the interaction between the total body and the flexible body dynamics, steady state maneuvers (i.e. constant angular rates of the body reference frame) are studied.

By putting the derivative of ω to zero and including the aerodynamic force representation from equation (41), equation (43) becomes:

$$\begin{bmatrix} I & 0 & 0 & 0 \\ \frac{-\rho V}{2}B_{Z0} & M' & \frac{-\rho V}{2}B_{Z1} & \frac{-\rho V}{2}B_{Z2} \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \begin{Bmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \\ \dot{\eta}_3 \\ \dot{\eta}_4 \end{Bmatrix} + \begin{bmatrix} 0 & -I & 0 & 0 \\ K'+A'-\frac{\rho V^2}{2}B_{T0} & G'+D' & \frac{-\rho V^2}{2}B_{T1} & \frac{-\rho V^2}{2}B_{T2} \\ 0 & -I & \beta_1 & 0 \\ 0 & -I & 0 & \beta_2 \end{bmatrix} \begin{Bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \eta_4 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad (59)$$

Note that the coefficient matrices on LHS of the equation (59) are time invariant when the angular rate, ω, is constant. Hence the eigenvalues of the system can be used to check the stability of the system and to study the effects on modes of the system at different angular rates.

6. APPLICATION TO FINITE ELEMENT MODEL

A large-order finite element model (FEM) of a generic fighter was obtained for use in the application of this method. The aircraft planform is similar to an F/A-18, although stiffness and mass data do not necessarily represent this airplane. Although the FEM consists primarily of beam elements, it is a highly detailed model containing an A-set of 228 degrees of freedom (DOF) and approximately 200 structural elements. Aerodynamic modeling of the aircraft consisted of 230 boxes, and can be seen in Figure 2. The doublet lattice method was used to formulate aerodynamic influence coefficient matrices. Eight values of reduced frequency were used to calculate unsteady aerodynamic matrices.

Certain assumptions used to develop the equations required that some modifications be made to the model. The equations assume that the mathematical model has six DOF for every subbody. If these matrices are generated from a FEM, this is rarely true. In NASTRAN , this corresponds to the initial global set (G-set) of coordinates. These DOF cannot normally be used, however, because many are constrained due to the method of modeling and imposition of boundary conditions.

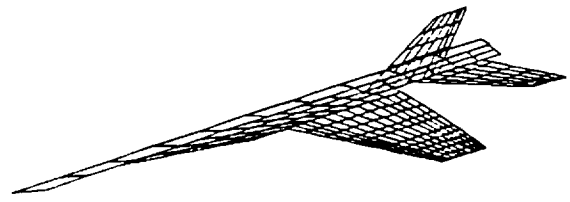


Figure 2. Aerodynamic Configuration

Table 3. Flexible Mode Frequency and Damping for Roll Rate Maneuvers
Full Stiffness

MSC/NASTRAN FLUTTER ANALYSIS		FLXAIR ANALYSIS 0.0 DEG/SEC ROLL RATE		FLXAIR ANALYSIS 90.0 DEG/SEC ROLL RATE		FLXAIR ANALYSIS 180.0 DEG/SEC ROLL RATE		FLXAIR ANALYSIS 240.0 DEG/SEC ROLL RATE	
FREQUENCY Hz	DAMPING	FREQUENCY Hz	DAMPING	FREQUENCY Hz	DAMPING	FREQUENCY Hz	DAMPING	FREQUENCY Hz	DAMPING
6.734	-.0689	6.699	-.0653	6.696	-.0653	6.685	-.0655	6.673	-.0656
8.948	-.000459	8.957	-.000422	8.955	-.000467	8.946	-.000539	8.936	-.000602
9.085	-.0740	9.056	-.0722	9.053	-.0722	9.044	-.0722	9.035	-.0722
14.192	-0.175	14.371	-0.142	14.368	-0.142	14.361	-0.142	14.353	-0.142
16.434	-.0730	16.779	-.0806	16.778	-.0806	16.774	-.0806	16.771	-.0805
18.736	-.0249	18.756	-.0297	18.755	-.0298	18.753	-.0298	18.751	-.0298
21.172	-.0188	21.812	-.0157	21.813	-.0159	21.814	-.0165	21.815	-.0162
23.172	-.0268	23.333	-.0292	23.333	-.0293	23.331	-.0294	23.333	-.0294
24.352	-.0406	24.692	-.0416	24.691	-.0416	24.689	-.0416	24.688	-.0416
29.578	-.00707	29.719	-.00878	29.276	-.00871	29.721	-.00865	29.722	-.00863
32.916	-.0458	33.439	-.0387	33.448	-.0387	33.437	-.0387	33.436	-.0387

These constrained DOF present a problem which requires either the modification of the equations or of the input matrices.

Another assumption made in the equations is that the mass matrix is block diagonal. However, the typical mass matrix from a FEM analysis contains coupling terms. These arise because of the following reasons:

1) Mass data may be input at locations other than structural grid point locations.
2) Coupling results from the use of dependency relations (multi-point constraints in NASTRAN).
3) Coupling results from the static reduction if inertia is lumped on any of the omitted DOF (Guyan reduction).

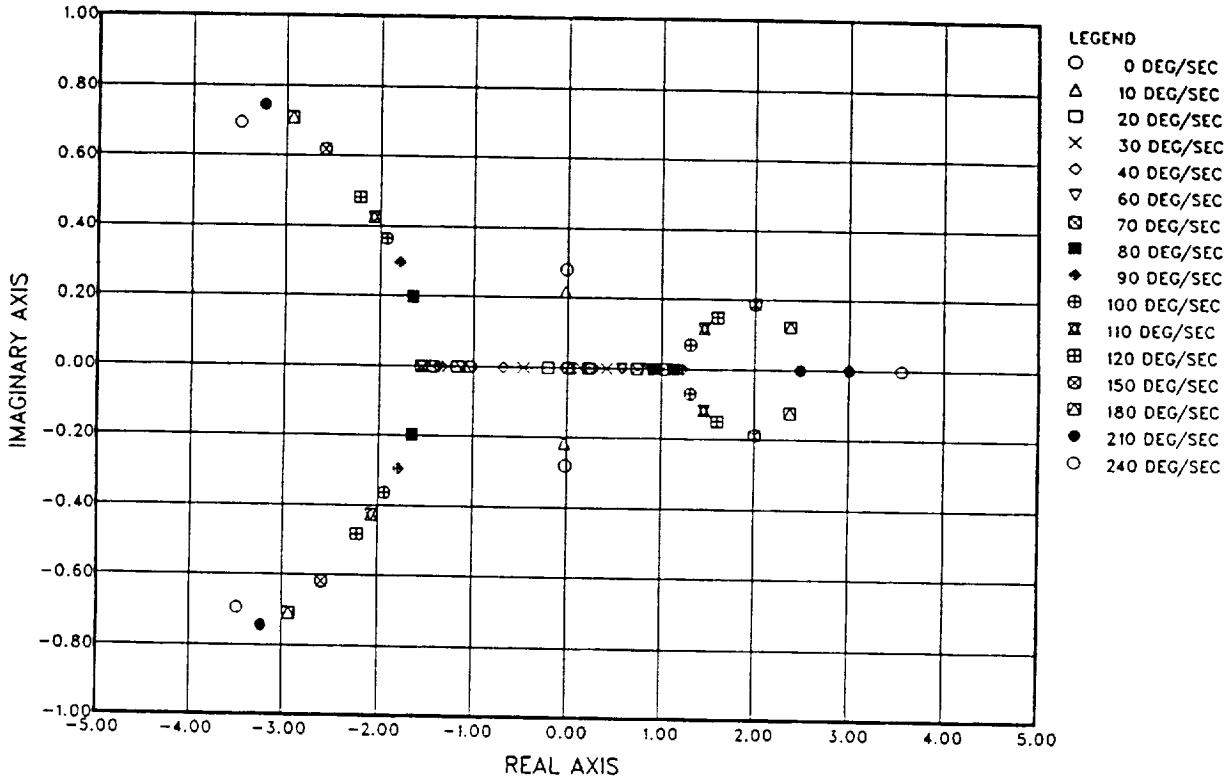
These considerations make it necessary to adjust the model as follows:

1) The inertia is relumped so that it is located at exact grid point locations.
2) Inertia at dependent DOF is relumped so that it is associated only with independent DOF.
3) Inertia located at DOF which are eliminated by the Guyan reduction process must be relumped at retained DOF (A-set).
4) A Boolean transformation matrix is formed for use in expansion of the FEM A-set DOF to the 6n DOF required by the equations. After forming the state space equations, this same matrix can be used to eliminate those DOF.

A NASTRAN flutter analysis of the vehicle was conducted for a case representing Mach .7 and an altitude of 20,000 ft. Mass, stiffness, and aerodynamic matrix data were obtained from NASTRAN for this case. The necessary matrices for the rational function approximation of the aerodynamics were obtained by a least squares fit using aerodynamic matrices for reduced frequencies of 0.0, 0.2, and 0.8 . The state space equations were formed and eigenvalue solutions were obtained for various values of roll rate.

For zero roll rate, the results agreed with the NASTRAN analysis. Increasing roll rate showed little effect on the flexible modes of the system, as can be seen in Table 3. The rigid body modes were affected, however. A root locus plot of the rigid body roots as a function of roll rate is shown in Figure 3. For zero roll rate, two stable real roots and one stable complex conjugate pair are obtained - corresponding to a roll convergence mode, a spiral mode, and an oscillatory dutch roll mode. With increasing roll rate, however, we see

Figure 3. Rigid Body Eigenvalues for Roll Rate Maneuvers Full Stiffness



that some roots become unstable, and also change from real to complex and back again to real. Another case, representing a more flexible airplane, shows the same behavior (Figure 4), although the changes occur at lower roll rates. This case represents 50% of the initial overall airplane stiffness. Table 4 shows again that the flexible modes were not greatly affected, even for the reduced stiffness case.

7. DISCUSSION

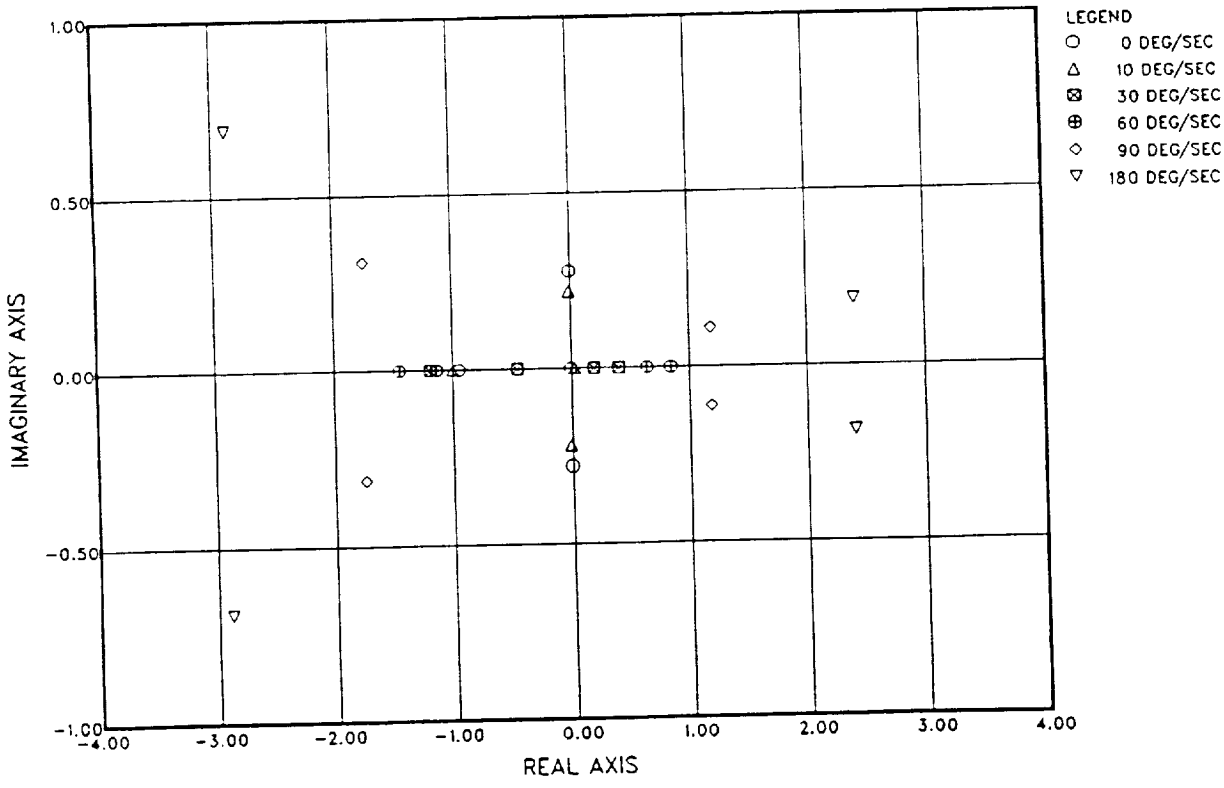
Dynamic equations have been derived for a flexible fixed wing aircraft, including an explicit representation of unsteady aerodynamic forces. The aircraft is assumed to be a collection of elastically interconnected discrete rigid subbodies. Deformations are assumed to be small in the body reference frame, thus allowing the equations to be linearized in the deformation variables.

The hybrid coordinate method is used to derive the total body and the elemental body dynamic equations which are then converted to matrix form. These equations are integrated in a state space format, along with a rational function approximation of the unsteady aerodynamic forces. These equations can be used for simulation. For the case of constant angular velocities of the body reference frame, the coefficient matrices become time invariant, allowing the use of an eigenvalue analysis to evaluate the effects of the angular rates on the system dynamic properties. When this method is applied to a realistic finite element model of a generic high-performance fighter, significant changes in the stability characteristics of the aircraft are observed. With increasing roll rate, some roots become unstable, and also change back and forth from complex to real. The dutch roll mode becomes two real roots, one of which combines with the spiral mode to produce an unstable oscillatory mode. The other real root from the original dutch roll mode

Table 4. Flexible Mode Frequency and Damping for Roll Rate Maneuvers
50 Percent Stiffness

FLXAIR ANALYSIS 0.0 DEG/SEC ROLL RATE		FLXAIR ANALYSIS 60.0 DEG/SEC ROLL RATE		FLXAIR ANALYSIS 90.0 DEG/SEC ROLL RATE		FLXAIR ANALYSIS 180.0 DEG/SEC ROLL RATE	
FREQUENCY Hz	DAMPING	FREQUENCY Hz	DAMPING	FREQUENCY Hz	DAMPING	FREQUENCY Hz	DAMPING
4.860	-.07367	4.858	-.0737	4.855	-.0737	4.839	-.0739
6.335	-.000492	6.333	-.000553	6.331	-.000590	6.318	-.000727
6.500	-.0929	6.500	-.0930	6.500	-.0930	6.484	-.0930
10.230	-0.186	10.229	-0.186	10.227	-0.186	10.216	-0.185
11.871	-.0873	11.871	-.0874	11.870	-.0874	11.867	-.0873
13.334	-.0431	13.334	-.0431	13.333	-.0432	13.331	-.0431
15.435	-.0196	15.435	-.0199	15.435	-.0200	15.436	-.0204
16.527	-.0364	16.527	-.0365	16.527	-.0366	16.525	-.0367
17.498	-.0578	17.497	-.0578	17.497	-.0578	17.494	-.0578
21.021	-.0125	21.021	-.0124	21.022	-.0124	21.024	-.0122
23.671	-.0479	23.711	-.0479	23.671	-.0479	23.669	-.0479

Figure 4. Rigid Body Eigenvalues for Roll Rate Maneuvers
50 Percent Stiffness



combines with the roll convergence to form another oscillatory mode which becomes more stable with increasing roll rate. The effect on the flexible modes of the aircraft was minimal for this configuration. The behavior of the rigid body modes is somewhat dependent on airframe stiffness, as can be observed for the 50% stiffness case.

It is expected that a design with increased span would show a greater effect due to roll rate for both the rigid body and flexible modes. This should be given consideration in the design of any future high-performance aircraft.

REFERENCES

[1] Meirovitch, L. and Quinn, R.D., "Equations of Motion for Maneuvering Flexible Spacecraft," Journal of Guidance, Control, and Dynamics, Vol. 10, Sept.-Oct. 1987, pp. 453-465.

[2] Laskin, R.A. and Likins, P.W., "Dynamical Equations of a Free-Free Beam Subject to Large Overall Motions," Proceedings of AAS/AIAA Astrodynamics Specialist Conference, Lake Tahoe, Nevada, Aug. 1981, paper 81-119

[3] Kane, T.R. and Levinson, D.A., "Formulation of Equations of Motion for Complex Spacecraft," Journal of Guidance and Control, Vol. 3, Mar.-Apr. 1980, pp. 99-112.

[4] Kane, T.R. and Levinson, D.A., "Large Motions of Unrestrained Space Trusses," The Journal of the Astronautical Sciences, Vol. XXVIII, Jan.-Mar., 1980, pp. 49-88.

[5] Likins, P.W., "Dynamics and Control of Flexible Space Vehicles," Jet Propulsion Laboratory, Pasadena, California, Technical Report 32-1329, Revision 1, Jan. 15, 1970.

[6] Likins, P.W., "Finite Element Appendage Equations for Hybrid Coordinate Dynamic Analysis," Jet Propulsion Laboratory, Pasadena, California, Technical Report 32-1525, Oct. 15, 1971.

[7] H.M. Youssef, A.P. Nayak, K.G. Gousman, "Integrated Total and Flexible Body Dynamics of Fixed Wing Aircraft," AIAA paper No. 88-2364 presented at 29th SDM Conference, Williamsburg, Virginia, April 18-20,1988.

[8] Roger, K. L., " Airplane Math Modeling Methods for Active Control Design," in AGARD Structures and Materials Panel, 44th Meeting, Lisbon, Portugal, April 1977.

A RECURSIVE APPROACH TO THE EQUATIONS OF MOTION FOR THE MANEUVERING AND CONTROL OF FLEXIBLE MULTI-BODY SYSTEMS

Moon K. Kwak and Leonard Meirovitch

Department of Engineering Science and Mechanics

Virginia Polytechnic Institute and State University

Blacksburg, VA 24061

157

Work supported in part by the AFOSR Research Grant F49620-89-C-0049DEF monitored by Dr. Spencer T. Wu; the support is greatly appreciated.

OVERVIEW

- The interest lies in a mathematical formulation capable of accommodating the problem of maneuvering a space structure consisting of a chain of articulated flexible substructures.

- Simultaneously, any perturbations from the "rigid-body" maneuvering and any elastic vibration must be suppressed.

- The equations of motion for flexible bodies undergoing rigid-body motions and elastic vibrations can be obtained conveniently by means of Lagrange's equations in terms of quasi-coordinates.

- The advantage of this approach is that it yields equations in terms of body axes, which are the same axes that are used to express the control forces and torques.

OVERVIEW (CONT'D)

- The equations of motion are nonlinear hybrid (ordinary and partial) differential equations.
- The partial differential equations can be discretized (in space) by means of the finite element method or the classical Rayleigh-Ritz method.
- The result is a set of nonlinear ordinary differential equations of high order.
- The nonlinearity can be traced to the rigid-body motions and the high order to the elastic vibration.
- Elastic motions tend to be small when compared with rigid-body motions.
- A perturbation approach permits breaking the problem into one for the rigid-body motions, which is nonlinear and of relatively low order, and for the elastic motions and the perturbations caused by them, which is linear and of relatively high order.

OVERVIEW (CONT'D)

- The rigid-body problem, which is associated with the maneuvering, is referred to as the zero-order (in a perturbation sense) problem and the control tends to be open loop.

- The perturbation suppression, which is associated with control, is referred to as the first-order problem and the control is closed-loop.

- The equations of motion are first derived for each individual substructure and then assembled into a single set for the fully interacting structure.

- The above is carried out by means of a kinematical synthesis eliminating the surplus coordinates.

- The kinematical synthesis, based on recursive relations, is carried out both for the zero-order and first-order problems.

HYBRID EQUATIONS FOR THE SUBSTRUCTURES

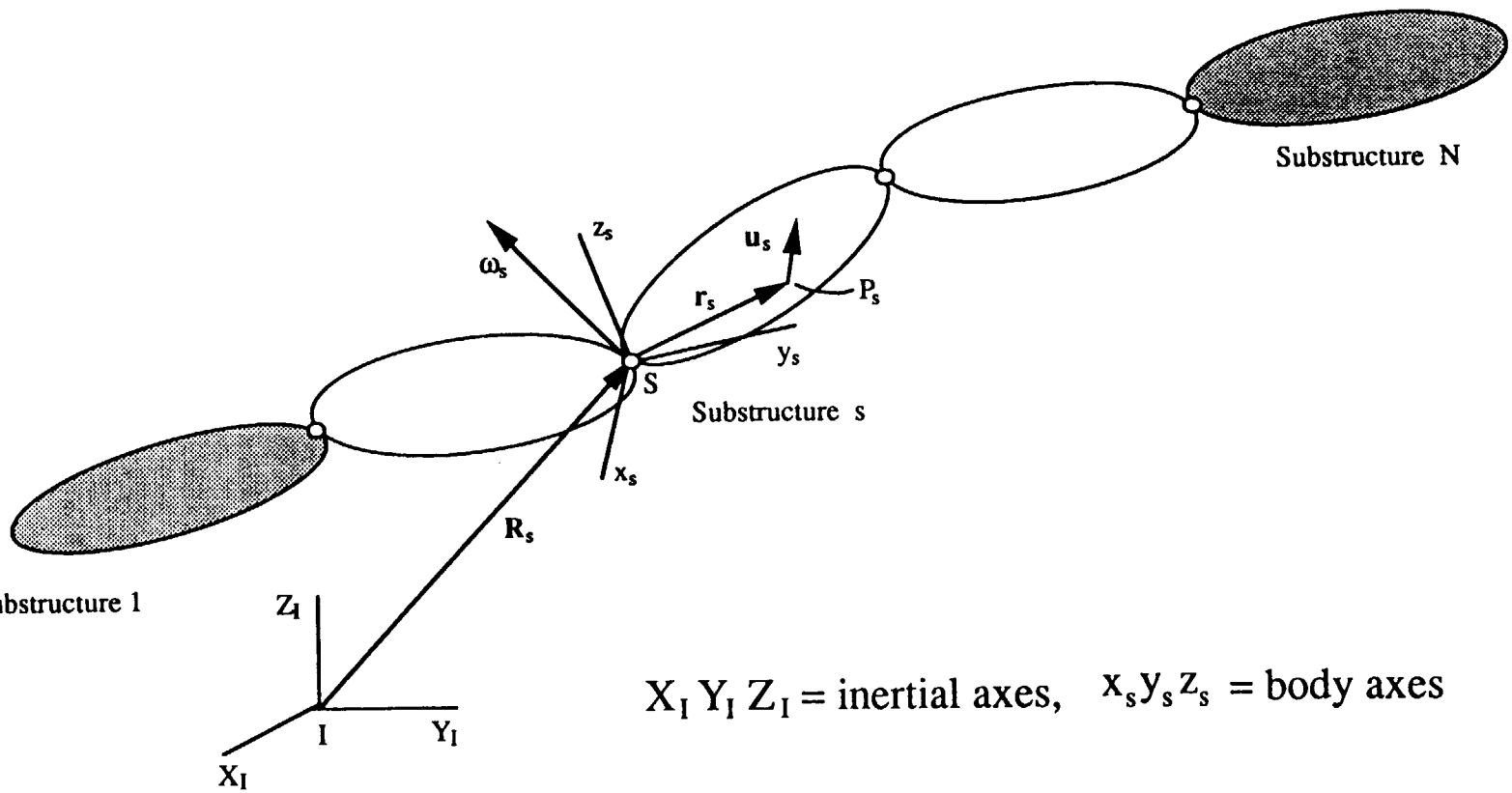


Figure 1 - The Articulated Chain of Substructures

HYBRID EQUATIONS FOR THE SUBSTRUCTURES

Derivation of the equations of motion by the Lagrangian approach requires the Lagrangian, and hence the kinetic and potential energy.

Position Vector of Point P_s in Substructure s :

$$\mathbf{W}_s(t) = \mathbf{R}_s(t) + \mathbf{r}_s(P_s) + \mathbf{u}_s(P_s, t), \quad s = 1, 2, ..., N$$

$\mathbf{R}_s$ = radius vector from I to S; in terms of inertial coordinates
$\mathbf{r}_s$ = radius vector from S to P_s; in terms of body coordinates
$\mathbf{u}_s$ = elastic displacement vector of P_s ; in terms of body coordinates

Velocity Vector of P_s:

$$\begin{aligned} \dot{\mathbf{W}}_s(t) &= \mathbf{V}_s(t) + \tilde{\omega}_s(t)(\mathbf{r}_s(P_s) + \mathbf{u}_s(P_s, t)) + \mathbf{v}_s(P_s, t) \\ &= \mathbf{V}_s(t) + (\tilde{r}_s(P_s) + \tilde{u}_s(P_s, t))^T \omega_s(t) + \mathbf{v}_s(P_s, t), \quad s = 1, 2, \ldots, N \end{aligned}$$

$\mathbf{V}_s$ = velocity vector of S ; in terms of body coordinates
ω_s = absolute angular velocity vector of $x_s y_s z_s$; in terms of body coordinates
$\tilde{\omega}_s$ = skew symmetric matrix formed from ω_s
$\mathbf{v}_s = \dot{\mathbf{u}}_s$ = elastic velocity vector of P_s ; in terms of body coordinates

HYBRID EQUATIONS FOR THE SUBSTRUCTURES(CONT'D)

Relation Between Inertial and Body-Axes Velocity Vectors:

$$\mathbf{V}_s = C_s \dot{\mathbf{R}}_s \ , \quad \boldsymbol{\omega}_s = D_s \dot{\boldsymbol{\theta}}_s$$

$C_s = C_s(\theta_{s1}, \theta_{s2}, \theta_{s3})$ = matrix of direction cosines between $x_s y_s z_s$ and $X_I Y_I Z_I$

$D_s = D_s(\theta_{s1}, \theta_{s2}, \theta_{s3})$ = transformation matrix

$\theta_{s1}, \theta_{s2}, \theta_{s3}$ = angles defining the orientation of $x_s y_s z_s$ and referred to $X_I Y_I Z_I$

$\mathbf{V}_s$, ω_s can be regarded as time derivatives of quasi-coordinates

HYBRID EQUATIONS FOR THE SUBSTRUCTURES(CONT'D)

Kinetic Energy:

$$
\begin{aligned}
T_s &= \frac{1}{2}\int_{D_s} \rho_s \dot{\mathbf{W}}_s^T \dot{\mathbf{W}}_s dD_s \\
&= \frac{1}{2} m_s \mathbf{V}_s^T \mathbf{V}_s + \mathbf{V}_s^T \tilde{S}_s^T \boldsymbol{\omega}_s + \mathbf{V}_s^T \int_{D_s} \rho_s \mathbf{v}_s dD_s \\
&\quad + \frac{1}{2}\boldsymbol{\omega}_s^T J_s \boldsymbol{\omega}_s + \boldsymbol{\omega}_s^T \int_{D_s} \rho_s (\tilde{r}_s + \tilde{u}_s)\mathbf{v}_s dD_s + \frac{1}{2}\int_{D_s} \rho_s \mathbf{v}_s^T \mathbf{v}_s dD_s
\end{aligned}
$$

$$
m_s = \int_{D_s} \rho_s dD_s, \; \tilde{S}_s = \int_{D_s} \rho_s (\tilde{r}_s + \tilde{u}_s) dD_s, \; J_s = \int_{D_s} \rho_s (\tilde{r}_s + \tilde{u}_s)(\tilde{r}_s + \tilde{u}_s)^T dD_s
$$

Potential Energy:

$$
V_s = \frac{1}{2}[\mathbf{u}_s, \mathbf{u}_s]
$$

ρ_s = mass density; D_s = domain of substructure s

[,] = energy inner product

HYBRID EQUATIONS FOR THE SUBSTRUCTURES(CONT'D)

General Hybrid Lagrange's Equations in Terms of Quasi-Coordinates:

$$\frac{d}{dt}\left(\frac{\partial L_s}{\partial \mathbf{V}_s}\right)+\tilde{\omega}_s\left(\frac{\partial L_s}{\partial \mathbf{V}_s}\right)-C_s\left(\frac{\partial L_s}{\partial \mathbf{R}_s}\right)=\mathbf{F}_s$$

$$\frac{d}{dt}\left(\frac{\partial L_s}{\partial \omega_s}\right)+\tilde{V}_s\left(\frac{\partial L_s}{\partial \mathbf{V}_s}\right)+\tilde{\omega}_s\left(\frac{\partial L_s}{\partial \omega_s}\right)-(D_s^T)^{-1}\left(\frac{\partial L_s}{\partial \boldsymbol{\theta}_s}\right)=\mathbf{M}_s$$

$$\frac{\partial}{\partial t}\left(\frac{\partial \hat{L}_s}{\partial \mathbf{v}_s}\right)-\left(\frac{\partial \hat{T}_s}{\partial \mathbf{u}_s}\right)+\mathcal{L}_s\mathbf{u}_s=\hat{\mathbf{U}}_s$$

$L_s=T_s-V_s$ = Lagrangian; $\hat{L}_s$ = Lagrangian density

$\hat{T}_s$ = kinetic energy density; $\mathcal{L}$ = (stiffness) differential operator matrix

$\mathbf{F}_s$, $\mathbf{M}_s$ = resultant force and torque vectors

$\hat{\mathbf{U}}_s$ = force density vector

HYBRID EQUATIONS FOR THE SUBSTRUCTURES(CONT'D)

Explicit Hybrid Equations:

$$m_s\dot{\mathbf{V}}_s + \tilde{S}_s^T\dot{\omega}_s + \int_{D_s}\rho_s\dot{\mathbf{v}}_s dD_s = (2\tilde{S}_{vs} + m_s\tilde{V}_s + \tilde{\omega}_s\tilde{S}_s)\omega_s + C_s\left(\frac{\partial L_s}{\partial \mathbf{R}_s}\right) + \mathbf{F}_s$$

$$\tilde{S}_s\dot{\mathbf{V}}_s + J_s\dot{\omega}_s + \int_{D_s}\rho_s(\tilde{r}_s+\tilde{u}_s)\dot{\mathbf{v}}_s dD_s = (\tilde{S}_s\tilde{V}_s - \tilde{\omega}_s J_s - J_{vs})\omega_s - \tilde{\omega}_s\int_{D_s}\rho_s(\tilde{r}_s+\tilde{u}_s)\mathbf{v}_s dD_s + (D_s^T)^{-1}\left(\frac{\partial L_s}{\partial \theta_s}\right) + \mathrm{M}_s$$

$$\rho_s[\dot{\mathbf{V}}_s + (\tilde{r}_s+\tilde{u}_s)^T\dot{\omega}_s + \dot{\mathbf{v}}_s] = \rho_s(\tilde{V}_s + 2\tilde{v})\omega_s - \rho_s\tilde{\omega}_s^2(\mathbf{r}_s+\mathbf{u}_s) - \mathcal{L}_s\mathbf{u}_s + \hat{\mathbf{U}}_s$$

where

$$S_{vs} = \dot{S}_s = \int_{D_s}\rho_s\tilde{v}_s dD_s\ ,\quad J_{vs} = \dot{J}_s = \int_{D_s}\rho_s[\tilde{v}_s(\tilde{r}_s+\tilde{u}_s)^T + (\tilde{r}_s+\tilde{u}_s)\tilde{v}_s^T]dD_s$$

166

Augmenting Equations:

$$\dot{\mathbf{R}}_s = C_s^T\mathbf{V}_s\ ,\quad \dot{\boldsymbol{\theta}}_s = D_s^{-1}\boldsymbol{\omega}_s\ ,\quad \dot{\mathbf{u}}_s = \mathbf{v}_s$$

ORDINARY DIFFERENTIAL EQUATIONS

Elastic Displacement Vector: $\mathbf{u}_s(P_s,t) = \Phi_s(P_s)\mathbf{q}_s(t), s = 1,2,\ldots,N$

Φ_s = matrix of admissible functions (shape functions)

$\mathbf{q}_s$ = vector of generalized displacements

- Derive discretized T_s and V_s

Discretized State Equations:

$$m_s\dot{\mathbf{V}}_s + \tilde{S}_{s0}^T\dot{\omega}_s + \bar{\Phi}_s\dot{\mathbf{p}}_s = -m_s\tilde{\omega}_s\mathbf{V}_s - \tilde{\omega}_s\tilde{S}_{s0}^T\omega_s - 2\tilde{\omega}_s\bar{\Phi}_s\mathbf{p}_s - (\tilde{\dot{\omega}}_s + \tilde{\omega}_s^2)\bar{\Phi}_s\mathbf{q}_s + \mathbf{F}_s$$

$$\tilde{S}_{s0}\dot{\mathbf{V}}_s + J_{s0}\dot{\omega}_s + \tilde{\Phi}_s\dot{\mathbf{p}}_s = -\tilde{S}_{s0}\tilde{\omega}_s\mathbf{V}_s - \tilde{\omega}_s J_{s0}\omega_s - 2\hat{\Phi}_s\mathbf{p}_s - [(\widetilde{[\tilde{V}_s\omega_s]} - \dot{\tilde{V}}_s)\bar{\Phi}_s + 2\dot{\hat{\Phi}}_s + 2\tilde{\omega}_s\hat{\Phi}_s - (\tilde{\dot{\omega}}_s + \tilde{\omega}_s^2)\tilde{\Phi}_s]\mathbf{q}_s + \mathbf{M}_s$$

$$\bar{\Phi}_s^T\dot{\mathbf{V}}_s + \tilde{\Phi}_s^T\dot{\omega}_s + M_s\dot{\mathbf{p}}_s = -\bar{\Phi}_s^T\tilde{\omega}_s\mathbf{V}_s + \hat{\Phi}_s^T\omega_s - 2\tilde{H}_s\mathbf{p}_s - [K_s + \bar{H}_s(\omega_s) + \dot{H}_s)]\mathbf{q}_s + \mathbf{Q}_s$$

Various terms involve integrals over D_s

Augmenting Equations: $\dot{\mathbf{R}}_s = C_s^T\mathbf{V}_s\ ,\ \dot{\theta}_s = D_s^{-1}\omega_s\ ,\ \dot{\mathbf{q}}_s = \mathbf{p}_s$

PERTURBATION APPROACH

Perturbation Expansions:

$$\mathbf{V}_s = \mathbf{V}_{s0} + \mathbf{V}_{s1} \ , \ \boldsymbol{\omega}_s = \boldsymbol{\omega}_{s0} + \boldsymbol{\omega}_{s1} \ , \ \mathbf{F}_s = \mathbf{F}_{s0} + \mathbf{F}_{s1} \ , \ \mathbf{M}_s = \mathbf{M}_{s0} + \mathbf{M}_{s1}$$

- Subscript 0 denotes zero-order (in a perturbation sense) quantities
- Subscript 1 denotes first-order quantities
- First-order terms are one order of magnitude smaller than zero-order terms
- Elastic displacements and velocities are by definition of first order.

PERTURBATION APPROACH(CONT'D)

Introduce perturbation expansion into state equations and separate orders of magnitude.

Zero-Order State Equations: $\mathcal{M}_{s0}\dot{\mathbf{x}}_{s0} = \mathcal{C}_{s0}\mathbf{x}_{s0} + \mathcal{B}_{s0}\mathbf{f}_{s0} + \mathcal{D}_{s0}\mathbf{d}_{s0}$

Zero-Order State and Excitation Vectors:

$$\mathbf{x}_{s0}(t) = [\mathbf{R}_{s0}^T(t)\ \boldsymbol{\theta}_{s0}^T(t)\ \mathbf{V}_{s0}^T(t)\ \boldsymbol{\omega}_{s0}^T(t)]^T\ ,\quad \mathbf{f}_{s0}(t) = [\mathbf{F}_{s0}^T(t)\ \mathbf{M}_{s0}^T(t)]^T$$

Coefficient Matrices:

$$\mathcal{M}_{s0} = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & m_s I & \tilde{S}_{s0}^T \\ 0 & 0 & \tilde{S}_{s0} & J_{s0} \end{bmatrix},\quad \mathcal{C}_{s0} = \begin{bmatrix} 0 & 0 & C_{s0}^T & 0 \\ 0 & 0 & 0 & D_{s0}^{-1} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathcal{B}_{s0} = \mathcal{D}_{s0} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ I & 0 \\ 0 & I \end{bmatrix},\quad \mathbf{d}_{s0} = \begin{bmatrix} -m_s\tilde{\omega}_{s0}\mathbf{V}_{s0} - \tilde{\omega}_{s0}\tilde{S}_{s0}^T\omega_{s0} \\ -\tilde{S}_{s0}\tilde{\omega}_{s0}\mathbf{V}_{s0} - \tilde{\omega}_{s0}J_{s0}\omega_{s0} \end{bmatrix}$$

PERTURBATION APPROACH(CONT'D)

First-Order State Equations: $\mathcal{M}_{s1}\dot{\mathbf{x}}_{s1} = \mathcal{C}_{s1}\mathbf{x}_{s1} + \mathcal{B}_{s1}\mathbf{f}_{s1} + \mathcal{D}_{s1}\mathbf{d}_{s1}$

First-Order State and Excitation Vectors:

$$\mathbf{x}_{s1}(t) = [\mathbf{U}_{s1}^T(t)\ \boldsymbol{\beta}_{s1}^T(t)\ \mathbf{q}_s^T(t)\ \mathbf{V}_{s1}^T(t)\ \boldsymbol{\omega}_{s1}^T(t)\ \mathbf{p}_s^T(t)]^T\ ,\quad \mathbf{f}_{s1}(t) = [\mathbf{F}_{s1}^T(t)\ \mathbf{M}_{s1}^T(t)\ \mathbf{Q}_s^T(t)]^T$$

$\mathbf{U}_{s1}$ = body-axes vector of perturbations in translational displacements

β_{s1} = body-axes vectors of perturbations in rotational displacements

Coefficient Matrices:

$$\mathcal{M}_{s1} = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & m_s I & \tilde{S}_{s0}^T & \bar{\Phi}_s \\ 0 & 0 & 0 & \tilde{S}_{s0} & J_{s0} & \tilde{\Phi}_s \\ 0 & 0 & 0 & \bar{\Phi}_s^T & \tilde{\Phi}_s^T & M_s \end{bmatrix}$$

PERTURBATION APPROACH(CONT'D)

Coefficient Matrices:

$$C_{s1} = \begin{bmatrix} -\tilde{\omega}_{s0} & -\tilde{V}_{s0} & 0 & I & 0 & 0 \\ 0 & -\tilde{\omega}_{s0} & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 & I \\ 0 & 0 & -(\dot{\tilde{\omega}}_{s0} + \tilde{\omega}_{s0}^2)\bar{\Phi}_{s0} & -m_s\tilde{\omega}_{s0} & -\Gamma_s & -2\tilde{\omega}_{s0}\bar{\Phi}_s \\ 0 & 0 & -\Xi_s & -\tilde{S}_s\tilde{\omega}_{s0} & -\Delta_s & -2\hat{\Phi}_s \\ 0 & 0 & -\bar{K}_s & -\bar{\Phi}_s^T\tilde{\omega}_{s0} & -\Upsilon_s & -2\bar{H}_s \end{bmatrix}$$

$$B_{s1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}, \quad \mathcal{D}_{s1} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ I \end{bmatrix}$$

in which

$$\Gamma_s = \tilde{S}_{s0}\tilde{\omega}_{s0} - 2\tilde{\omega}_{s0}\tilde{S}_{s0} - m_s\tilde{V}_{s0}\ , \quad \Delta_s = 2\tilde{\omega}_{s0}J_{s0} + J_{s0}\tilde{\omega}_{s0} - (trJ_{s0})\tilde{\omega}_{s0} - \tilde{S}_{s0}\tilde{V}_{s0}$$

$$\Xi_s = (\widetilde{[\tilde{V}_{s0}\boldsymbol{\omega}_{s0}]} - \dot{\tilde{V}}_{s0})\bar{\Phi}_s + 2\dot{\hat{\Phi}}_s + 2\tilde{\omega}_s\hat{\Phi}_s - (\dot{\tilde{\omega}}_{s0} + \tilde{\omega}_{s0}^2)\bar{\Phi}_s$$

$$\Upsilon_s = \bar{\Phi}_s^T\tilde{\omega}_{s0}^T - 2\hat{\Phi}_s^T + \bar{\Phi}_s^T\tilde{V}_{s0}^T, \ \bar{K}_s = K_s + \bar{H}_s + \dot{\bar{H}}_s$$

$$\mathbf{d}_{s1} = -\bar{\Phi}_s^T(\dot{\mathbf{V}}_{s0} + \tilde{\omega}_{s0}\mathbf{V}_{s0}) - \tilde{\Phi}_s\dot{\omega}_{s0} - \hat{\Phi}_s^T\omega_{s0}$$

KINEMATICAL SYNTHESIS FOR ZERO-ORDER EQUATIONS

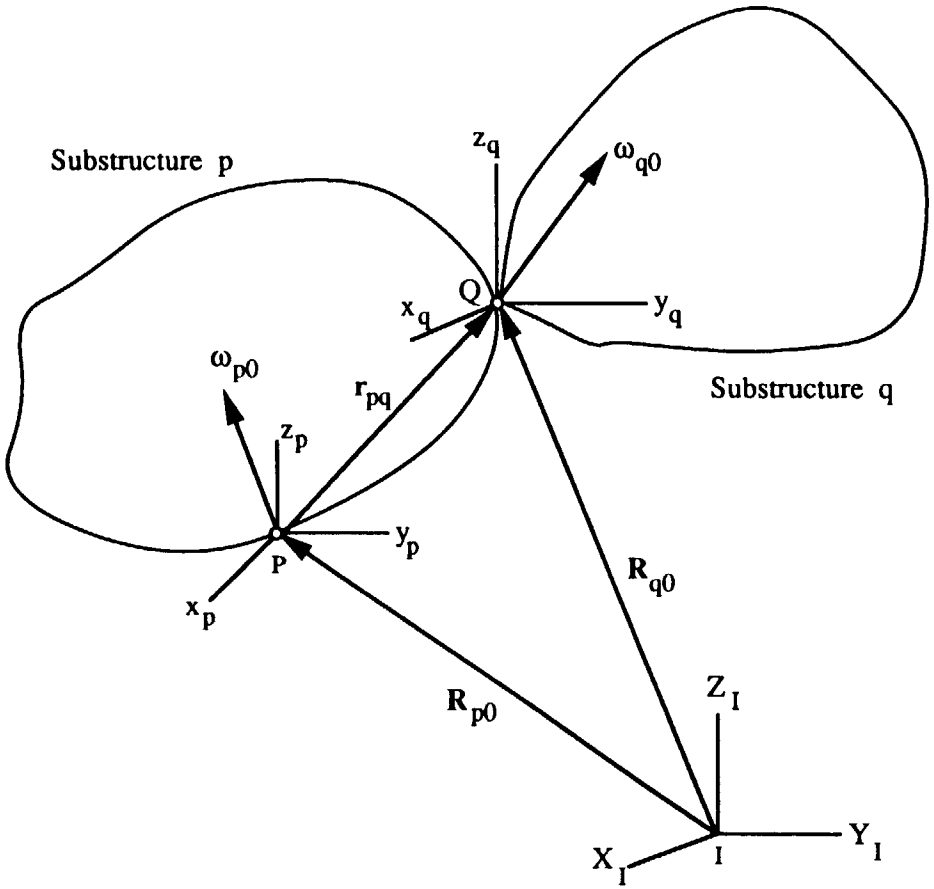


Figure 2 - Two Adjacent Substructures in the Chain

KINEMATICAL SYNTHESIS FOR ZERO-ORDER EQUATIONS (CONT'D)

Kinematical Constraints (linking the substructure together):

$$\mathbf{R}_{q0} = \mathbf{R}_{p0} + C_{p0}^T \mathbf{r}_{pq} \ , \quad \boldsymbol{\theta}_{q0} = \boldsymbol{\theta}_{q0}$$

$$\mathbf{V}_{q0} = C_{qp} \mathbf{V}_{p0} - C_{qp} \tilde{r}_{pq} \boldsymbol{\omega}_{p0} \ , \quad \boldsymbol{\omega}_{q0} = \boldsymbol{\omega}_{q0}$$

Recursive Relations:

$$\mathbf{R}_{s0} = \mathbf{R}_{10} + \sum_{j=2}^{s} C_{j-1,0}^T \mathbf{r}_{j-1,j}$$

$$\mathbf{V}_{s0} = \prod_{j=s}^{2} C_{j,j-1} \mathbf{V}_{10} - \prod_{j=s}^{2} C_{j,j-1} \tilde{r}_{12} \omega_{10} - \prod_{j=s}^{3} C_{j,j-1} \tilde{r}_{23} \omega_{20} \cdots - C_{s,s-1} \tilde{r}_{s-1,s} \omega_{s0}$$

Relation Between the State of Substructure s and Part of Constrained State of Structure (Substructures 1 through s) :

$$\mathbf{x}_{s0}^u = T_{s0} \mathbf{x}_{s0}^c + \bar{C}_s \mathbf{r}_{s0} \ , \quad s = 1, 2, \cdots, N$$

$$\mathbf{x}_{s0}^u = \mathbf{x}_{s0} \, , \ \mathbf{x}_{s0}^c = [\mathbf{R}_{10}^T \ \boldsymbol{\theta}_{10}^T \ \boldsymbol{\theta}_{20}^T \ \cdots \ \boldsymbol{\theta}_{s0}^T \ \mathbf{V}_{10}^T \ \boldsymbol{\omega}_{10}^T \ \boldsymbol{\omega}_{20}^T \ \cdots \ \boldsymbol{\omega}_{s0}^T]^T \ , \quad s = 1, 2, \cdots, N$$

$$\mathbf{r}_{s0} = [\mathbf{r}_{12}^T \ \mathbf{r}_{23}^T \ \cdots \ \mathbf{r}_{s-1,s}^T]^T \ , \quad s = 1, 2, \cdots, N$$

KINEMATICAL SYNTHESIS FOR ZERO-ORDER EQUATIONS (CONT'D)

$$\mathcal{M}_0\dot{\mathbf{x}}_0^u = \mathcal{C}_0\mathbf{x}_0^u + \mathcal{B}_0\mathbf{f}_0 + \mathcal{D}_0\mathbf{d}_0$$

Disjoint State Equations:

$$\mathbf{x}_0^u = \begin{bmatrix} \mathbf{x}_{10} \\ \mathbf{x}_{20} \\ \vdots \\ \mathbf{x}_{N0} \end{bmatrix}, \quad \mathbf{f}_0 = \begin{bmatrix} \mathbf{f}_{10} \\ \mathbf{f}_{20} \\ \vdots \\ \mathbf{f}_{N0} \end{bmatrix}, \quad \mathbf{d}_0 = \begin{bmatrix} \mathbf{d}_{10} \\ \mathbf{d}_{20} \\ \vdots \\ \mathbf{d}_{N0} \end{bmatrix}$$

Completion of the State Dimension:

$$\mathbf{x}_{10}^* = \begin{bmatrix} \mathbf{x}_{10} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{x}_{20}^* = \begin{bmatrix} 0 \\ \mathbf{x}_{20} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \cdots, \quad \mathbf{x}_{N0}^* = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \mathbf{x}_{N0} \end{bmatrix}$$

Unconstrained State: $\mathbf{x}_0^u = \sum_{s=1}^{N} \mathbf{x}_{s0}^*$

Full-Dimension Constraint Equation:

$$\mathbf{x}_0^u = \sum_{s=1}^{N} \mathbf{x}_{s0}^* = \sum_{s=1}^{N} T_{s0}^* \mathbf{x}_0^c + \bar{C}\mathbf{r}_0 = T_0\mathbf{x}_0^c + \bar{C}\mathbf{r}_0$$

Constrained State:

$$\mathbf{x}_0^c = \mathbf{x}_{N0}^c = [\mathbf{R}_{10}^T\ \boldsymbol{\theta}_{10}^T\ \boldsymbol{\theta}_{20}^T\ \cdots\ \boldsymbol{\theta}_{N0}^T\ \mathbf{V}_{10}^T\ \boldsymbol{\omega}_{10}^T\ \boldsymbol{\omega}_{20}^T\ \cdots\ \boldsymbol{\omega}_{N0}^T]^T, \quad \mathbf{r}_0 = \mathbf{r}_{N0}$$

KINEMATICAL SYNTHESIS FOR ZERO-ORDER EQUATIONS (CONT'D)

State Equations for Zero-Order Problems:

$$\dot{\mathbf{x}}_0 = \mathcal{A}_0\mathbf{x}_0 + \mathcal{B}_0^*\mathbf{f}_0 + \mathcal{D}_0^*\mathbf{d}_0 + \mathcal{R}_0^*\mathbf{r}_0$$

$$\mathcal{A}_0 = (T_0^T\mathcal{M}_0T_0)^{-1}T_0^T(\mathcal{C}_0T_0 - \mathcal{M}_0\dot{T}_0)\ ,\quad \mathcal{B}_0^* = (T_0^T\mathcal{M}_0T_0)^{-1}T_0^T\mathcal{B}_0$$

$$\mathcal{D}_0^* = (T_0^T\mathcal{M}_0T_0)^{-1}T_0^T\mathcal{D}_0\ ,\quad \mathcal{R}_0^* = (T_0^T\mathcal{M}_0T_0)^{-1}T_0^T(\mathcal{C}_0\bar{C} - \mathcal{M}_0\dot{\bar{C}})$$

Note: Superscript c was dropped for simplicity

KINEMATICAL SYNTHESIS FOR FIRST-ORDER EQS .

Kinematical Constraints Yield Recursive Relations:

$$\mathbf{U}_{q1} = C_{qp}[\mathbf{U}_{p1} - \tilde{r}_{pq}\beta_{p1} + \Phi_{pq}\mathbf{q}_p] \,, \quad \beta_{q1} = C_{qp}(\beta_{p1} + \Psi_{pq}\mathbf{q}_p) \,, \quad \mathbf{q}_q = \mathbf{q}_q$$

$$\mathbf{V}_{q1} = [\tilde{V}_{p0}C_{qp} + C_{qp}(\widetilde{[\tilde{r}_{pq}\omega_{p0}]} - \tilde{V}_{p0})]\beta_{p1} + [\tilde{V}_{p0}C_{qp}\Psi_{pq} + C_{qp}\tilde{\omega}_{p0}\Phi_{pq}]\mathbf{q}_p + C_{qp}\mathbf{V}_{p1} - C_{qp}\tilde{r}_{pq}\omega_{p1} + C_{qp}\Phi_{pq}\mathbf{p}_p$$

$$\omega_{q1} = C_{qp}(\tilde{\omega}_{p0}\Psi_{pq}\mathbf{q}_p + \omega_{p1} + \Psi_{pq}\mathbf{p}_p) \,, \quad \mathbf{p}_q = \mathbf{p}_q$$

where

$$\Phi_{pq} = \Phi_p(\mathbf{r}_{pq}) \,, \quad \Psi_{pq} = \nabla \times \Phi_p(\mathbf{r}_{pq})$$

Matrix Form of Recursive Relations:

$$\mathbf{x}^u_{s1} = T_{s1}\mathbf{x}^c_{s1} \,, \quad s = 2, 3, \cdots, N$$

Disjoint Perturbation State Equations:

$$\mathcal{M}_1\dot{\mathbf{x}}^u_1 = \mathcal{C}_1\mathbf{x}^u_1 + \mathcal{B}_1\mathbf{f}_1 + \mathcal{D}_1\mathbf{d}_1$$

$$\mathbf{x}^u_1 = \begin{bmatrix} \mathbf{x}_{11} \\ \mathbf{x}_{21} \\ \vdots \\ \mathbf{x}_{N1} \end{bmatrix} \,, \quad \mathbf{f}_1 = \begin{bmatrix} \mathbf{f}_{11} \\ \mathbf{f}_{21} \\ \vdots \\ \mathbf{f}_{N1} \end{bmatrix} \,, \quad \mathbf{d}_1 = \begin{bmatrix} \mathbf{d}_{11} \\ \mathbf{d}_{21} \\ \vdots \\ \mathbf{d}_{N1} \end{bmatrix}$$

KINEMATICAL SYNTHESIS FOR FIRST-ORDER EQS. (CONT'D)

Completion of the Perturbation State Dimension:

$$\mathbf{x}_{11}^* = \begin{bmatrix} \mathbf{x}_{11} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{x}_{21}^* = \begin{bmatrix} 0 \\ \mathbf{x}_{21} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{x}_{31}^* = \begin{bmatrix} 0 \\ 0 \\ \mathbf{x}_{31} \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{x}_{N1}^* = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \mathbf{x}_{N1} \end{bmatrix}$$

Constrained Perturbation State Vector:

$$\mathbf{x}_1^c = [\mathbf{U}_{11}^T \; \boldsymbol{\beta}_{11}^T \; \mathbf{q}_1^T \; \mathbf{q}_2^T \; \cdots \; \mathbf{q}_N^T \; \mathbf{V}_{11}^T \; \boldsymbol{\omega}_{11}^T \; \mathbf{p}_1^T \; \mathbf{p}_2^T \; \cdots \; \mathbf{p}_N^T]^T$$

Full-Dimensional Constraint Equation:

$$\mathbf{x}_1^u = \sum_{s=1}^{N} \mathbf{x}_{s1}^* = \sum_{s=1}^{N} T_{s1}^* \mathbf{x}_1^c = T_1 \mathbf{x}_1^c$$

State Equations for First-Order Problem:

$$\dot{\mathbf{x}}_1 = \mathcal{A}_1 \mathbf{x}_1 + \mathcal{B}_1^* \mathbf{f}_1 + \mathcal{D}_1^* \mathbf{d}_1$$

$$\mathcal{A}_1 = (T_1^T \mathcal{M}_1 T_1)^{-1} T_1^T (\mathcal{C}_0 T_0 - \mathcal{M}_0 \dot{T}_0), \quad \mathcal{B}_1^* = (T_1^T \mathcal{M}_1 T_1)^{-1} T_1^T \mathcal{B}_1$$

$$\mathcal{D}_1^* = (T_1^T \mathcal{M}_1 T_1)^{-1} T_1^T \mathcal{D}_1$$

SUMMARY AND CONCLUSIONS

- The equations of motion for a structure in the form of a collection of articulated flexible substructures can be derived conveniently by means of Lagrange's equations in terms of quasi-coordinates for flexible bodies.

- For practical reasons, the set of nonlinear hybrid (ordinary and partial) differential equations is transformed into a set of nonlinear ode's of high dimension.

- Due to the nature of the problem, a perturbation approach can be used to divide the equations into two sets containing terms differing in magnitude.

- The zero-order problem is nonlinear and of relatively low order. It is associated with the "rigid-body" maneuvering and the control is open loop.

SUMMARY AND CONCLUSIONS (CONT'D)

- The first-order problem is linear and of relatively high order. It is associated with the elastic vibration and the perturbation it causes in the rigid-body motions and the control is closed loop.

- The equations of motion are derived for each substructure separately.

- A given kinematical synthesis is used to link the various substructures together.

- The constraint equations lead to recursive relations that are used to eliminate the surplus coordinates.

- The procedure is used to derive state equations both for the zero-order and first-order problems.

- The formulation is particularly well suited for control design.

Serial and Parallel Computation of Kane's Equations for Multibody Dynamics

by

Amir Fijany

Jet Propulsion Laboratory, California Institute of Technology

Objective:

Analysis of the Efficiency of Algorithms resulting from Kane's Equation for Serial and Parallel Computation of Mass Matrix.

Overview:

* **Algorithms resulting from Kane's Equation and Modified Kane's Equation.**

* **Analysis of two Classes of Algorithms for Computation of Mass Matrix: The Newton-Euler Based Algorithms and the Composite Rigid-Body Algorithms.**

* **Analysis of the Efficiency of Different Algorithms for Serial and Parallel Computation.**

* **Conclusion**

Multibody Dynamics

Case Study: Rigid Multibody as Specialized to a Single Chain Robot Manipulator.

Multibody Dynamics: Solution for $\ddot{Q}$ from

$$A\ddot{Q} = \tau - b = \Gamma \qquad (1)$$

A: nxn Symmetric Positive definite Mass Matrix
$\ddot{Q}$: nx1 Vector of Generalized Accelerations
τ: nx1 Vector of Applied Forces/Torques
b: nx1 Vector of nonlinear Terms (Bias vector)
Γ: nx1 Vector of Applied Inertia Forces/Torques

The $O(n^3)$ Algorithms for Multibody Dynamics:

1) Computation of b and Γ.

2) Computation of Mass Matrix A.

3) Solution of Eq. (1) by Inversion of A.

Kane's Equation is widely used for Computation of Mass Matrix.

Kane's Method: Notation

Q: nx1 Vector of Generalized Coordinates

U: nx1 Vector of Generalized Speeds

Choice of U: $U_i = \sum_{j=1}^{n} A_{ij}\dot{Q}_j + B_i$

Angular and Linear Velocity of Body (Link) i

$$\underline{\omega}_i = \sum_{j=1}^{n} \underline{\omega}_{i(j)} U_j + \underline{\omega}_{i(t)}$$

$$\underline{V}^*_i = \sum_{j=1}^{n} \underline{V}^*_{i(j)} U_j + \underline{V}^*_{i(t)}$$

ω_i: Angular Velocity of Body i

$\omega_{i(j)}$: jth Partial Angular Velocity of Body i

$\omega_{i(t)}$: Angular Velocity of Remainder Terms

V^*_i: Linear Velocity of Center of Mass of Body i

$V^*_{i(j)}$: jth Partial Linear Velocity of Center of Mass of Body i

$V^*_{i(t)}$: Linear Velocity of Center of Mass of Body i Remainder Terms

Kane's Method: Notation

Partial Angular and Linear Momentum

$$\underline{N}_{i(j)} = \underline{\underline{I}}_i \underline{\omega}_{i(j)}$$

$$\underline{F}_{i(j)} = m_i \underline{V}_{i(j)}$$

$N_{i(j)}$: jth Partial Angular Momentum of Body i

$F_{i(j)}$: jth Partial Linear Momentum of Body i

Kane's Equation for Computation of Mass Matrix

The element a_{ij} of Mass Matrix A is Computed as

$$a_{ij} = \sum_{k=j}^{n} \underline{V}^*_{k(i)} \cdot \underline{F}_{k(j)} + \underline{\omega}_{k(i)} \cdot \underline{N}_{k(j)}$$

$$= \sum_{k=j}^{n} \underline{V}^*_{k(i)} \cdot m_k \underline{V}^*_{k(j)} + \underline{\omega}_{k(i)} \cdot \underline{\underline{I}}_k \underline{\omega}_{k(j)}$$

Kane's Equation: Analysis of General Case

For Analysis of the General Case, We Set $U_i = \dot{Q}_i$

$$\underline{\omega}_{i(j)} = \underline{Z}_j \text{ and } \underline{\omega}_{i(t)} = 0$$

$$\underline{\omega}_i = \sum_{j=1}^{i} \underline{Z}_j \dot{Q}_j$$

$$\underline{V}^*_{i(j)} = (\underline{Z}_j \times \underline{P}_{i*,j}) \text{ and } \underline{V}^*_{i(t)} = 0$$

$$\underline{V}^*_i = \sum_{j=1}^{i} (\underline{Z}_j \times \underline{P}_{i*,j}) \dot{Q}_j$$

$$\underline{N}_{i(j)} = \underline{\underline{I}}_i \underline{\omega}_{i(j)} = \underline{\underline{I}}_i \underline{Z}_j$$

$$\underline{F}_{i(j)} = m_i (\underline{Z}_j \times \underline{P}_{i*,j})$$

Kane's Equation can be written as

$$a_{ij} = \sum_{k=j}^{n} (\underline{Z}_i \times \underline{P}_{k*,i}) . m_k (\underline{Z}_j \times \underline{P}_{k*,j}) + \underline{Z}_i \cdot \underline{\underline{I}}_k \underline{Z}_j$$

AN O(n^3) Algorithm Based on Kane's Equation

For i = 1, 2, ..., n

For j = i, i+1, ..., n

$$a_{ij} = \sum_{k=j}^{n} (\underline{Z}_i \times \underline{P}_{k^*,i}) . m_k (\underline{Z}_j \times \underline{P}_{k^*,j}) + \underline{Z}_i . \underline{\underline{I}}_k \underline{Z}_j$$

This Algorithm is Designated as Original Kane's Equation (OKE) Algorithm.

Modified Kane's Equation

$$a_{ij} = \sum_{k=j}^{n} (\underline{Z}_j \times \underline{P}_{k^*,j}) . m_k (\underline{Z}_i \times \underline{P}_{k^*,i}) + \underline{Z}_j . \underline{\underline{I}}_k \underline{Z}_i$$

$$= \sum_{k=j}^{n} \underline{Z}_j . (\underline{P}_{k^*,j} \times (m_k \underline{Z}_i \times \underline{P}_{k^*,i}) + \underline{Z}_j . \underline{\underline{I}}_k \underline{Z}_i$$

$$= \sum_{k=j}^{n} \underline{Z}_j . ((\underline{P}_{k^*,j} \times (m_k \underline{Z}_i \times \underline{P}_{k^*,i}) + \underline{\underline{I}}_k \underline{Z}_i)$$

AN $O(n^2)$ Algorithm Based on Kane's Equation

For $i = 1, 2, \ldots, n$

For $j = i, i+1, \ldots, n$

$$\underline{P}_{j,i} = \underline{P}_{j-1,i} + \underline{P}_{j,j-1}$$

$$\underline{P}_{j^*,i} = \underline{P}_{j,i} + \underline{S}_j$$

$$\underline{N}_{j(i)} = \underline{\underline{I}}_j \underline{Z}_i$$

$$\underline{F}_{j(i)} = m_j(\underline{Z}_i \times \underline{P}_{j^*,i})$$

For $j = n, n-1, \ldots, i$

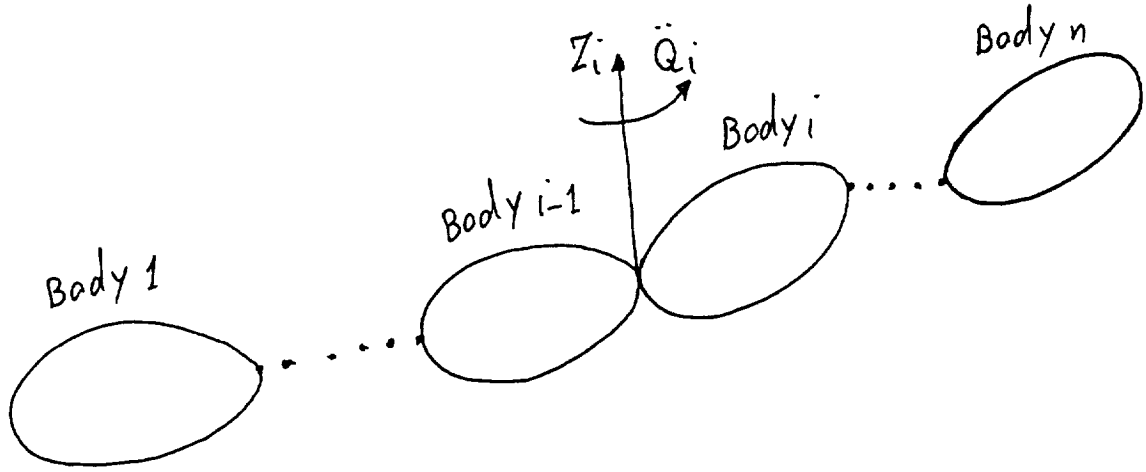
$$\underline{f}_{j(i)} = \underline{F}_{j(i)} + \underline{f}_{j+1(i)}$$

$$\underline{n}_{j(i)} = \underline{N}_{j(i)} + \underline{S}_j \times \underline{F}_{j(i)} + \underline{n}_{j+1(i)} + \underline{P}_{j+1,j} \times \underline{f}_{j+1(i)}$$

$$a_{ij} = \underline{Z}_j \cdot \underline{n}_{j(i)}$$

This Algorithm is Designated as Variant of Kane's Equation (VKE) Algorithm.

Algorithms for Computation of Mass Matrix



$$A\ddot{Q} = \Gamma \tag{1}$$

$$a_{ij} = a_{ji} = \Gamma_i \tag{2}$$

For the conditions given as

$$\ddot{Q}_i = 1 \text{ and } \dot{Q}_j = \ddot{Q}_{j \neq i} = 0 \quad \text{For } j = 1, 2, ..., n \tag{3}$$

Two physical interpretations of Eqs. (2) & (3) lead to two classes of algorithms for computation of mass matrix:

1. The Newton-Euler Based (N-E B) Algorithms.
   Underlying Physical Concept: Propagation of acceleration among rigidly connected bodies.
   The Variant of Kane's Equation (VKE) Algorithm belongs to this class.

2. The Composite Rigid-Body (CRB) Algorithms.
   Underlying physical Concept: Propagation of force among rigidly connected bodies.

Algorithms for Computation of Mass Matrix

Clearly, the two physical interpretations are the same. We have shown that the algorithms of the two classes can be transformed to one another.

From an algorithmic point of view, the main difference between the algorithms of the two classes is the presence of a two-dimensional recursion in Composite Rigid-Body Algorithms.

The main issue is to determine the best algorithm(s) for serial and parallel computation.

The Original Kane's Equation Algorithm is the least efficient since its computational complexity is of $O(n^3)$.

The computational complexity of both the Newton-Euler Based Algorithms and Composite Rigid-Body Algorithms is of $O(n^2)$. However, the Composite Rigid-Body Algorithms, in general, are more efficient.

Algorithms for Computation of Mass Matrix

There are four major redundancies in the Original Newton-Euler Based Algorithm which can be removed by:

1) Optimizing the Newton-Euler Formulation for the conditions given in Eq. (3),

2) Using a variant of Newton-Euler Formulation ,

3) Choosing a better coordinate frame for projection of equations.

4) Introducing a two-dimensional recursion in the computation which transforms it to an equivalent Composite Rigid-Body Algorithm.

A Variant of Newton-Euler Based Algorithm

Step 1:

For $j = 1, 2, \ldots, n$

For $i = j, j+1, \ldots, n$

$$\underline{\dot{\omega}}(i,j) = \underline{Z}(j)$$

$$\underline{\dot{V}}(i,j) = \underline{\dot{V}}(i-1,j) + \underline{\dot{\omega}}(i,j) \times \underline{P}(i,i-1)$$

$$\underline{F}(i+1,i,j) = m(i)\underline{\dot{V}}(i,j) + \underline{\dot{\omega}}(i,j) \times \underline{h}(i)$$

$$\underline{N}(i+1,i,j) = \underline{\underline{k}}(i)\underline{\dot{\omega}}(i,j)$$

Step 2:

For $i = n, n-1, \ldots, j$

$$\underline{F}(n+1,n+1,j) = \underline{N}(n+1,n+1,j) = 0$$

$$\underline{F}(n+1,i,j) = \underline{F}(i+1,i,j) + \underline{F}(n+1,i+1,j)$$

$$\underline{N}(n+1,i,j) = \underline{N}(i+1,i,j) + \underline{N}(n+1,i+1,j) + \underline{P}(i+1,i) \times \underline{F}(n+1,i+1,j)$$

$$a_{ji} = \underline{Z}(i) . \underline{N}(n+1,i+1,j)$$

This algorithm results from removing the first two redundancies of the O N-E B Algorithm. It is clearly equivalent to the $O(n^2)$ algorithm resulting from the Kane's Equation or the Variant of Kane's Equation (VKA) Algorithm.

m(i)	Mass of body i.
h(i)	First moment of mass of body i about point O_i.
k(i)	Second moment of mass of body i about point O_i.
Z(i)	Axis of joint i
P(i,j)	Position vector from point j to point i.
ω(i,j)	Angular acceleration of body i resulting from the unit acceleration of joint j.
V(i,j)	Linear acceleration of body i (point O_i) resulting from the unit acceleration of joint j.
F(k+1,i,j)	Force exerted on point O_i due to the acceleration of bodies i through k, i.e., the bodies contained between points O_i and O_{k+1}, resulting from the unit acceleration of joint j.
N(k+1,i,j)	Moment exerted on point O_i due to the acceleration of body i through k, resulting from the unit acceleration of joint j.

A Variant of Composite Rigid-Body Algorithm

Step 1:

For $i = n, n-1, \ldots, 1$

$$M(i) = m(i) + M(i+1)$$

$$\underline{H}(i) = \underline{h}(i) + \underline{H}(i+1) + M(i+1)\underline{P}(i+1,i)$$

$$\underline{\underline{K}}(i) = \underline{\underline{k}}(i) + \underline{\underline{K}}(i+1) - \underline{\underline{M}}(i+1)\hat{\underline{\underline{P}}}(i+1,i)\hat{\underline{\underline{P}}}(i+1,i) - \hat{\underline{\underline{P}}}(i+1,i)\hat{\underline{\underline{H}}}(i+1) - \hat{\underline{\underline{H}}}(i+1)\hat{\underline{\underline{P}}}(i+1,i)$$

$$\underline{f}(i) = \underline{Z}(i) \times \underline{H}(i)$$

$$\underline{n}(i) = \underline{\underline{K}}(i)\underline{Z}(i)$$

$$a_{ii} = \underline{Z}(i).\underline{n}(i)$$

Step 2:

For $j = i-1, i-2, \ldots, 1$

$$\underline{f}(j) = \underline{f}(j+1)$$

$$\underline{n}(j) = \underline{n}(j+1) + \underline{P}(j+1,j) \times \underline{f}(j+1)$$

$$a_{ji} = \underline{Z}(j).\underline{n}(j)$$

M(i)	Mass of composite rigid-body i composed of bodies i through n.
H(i)	First moment of mass of composite rigid-body i about point O_i.
K(i)	Second moment of mass of composite rigid-body i about point O_i.

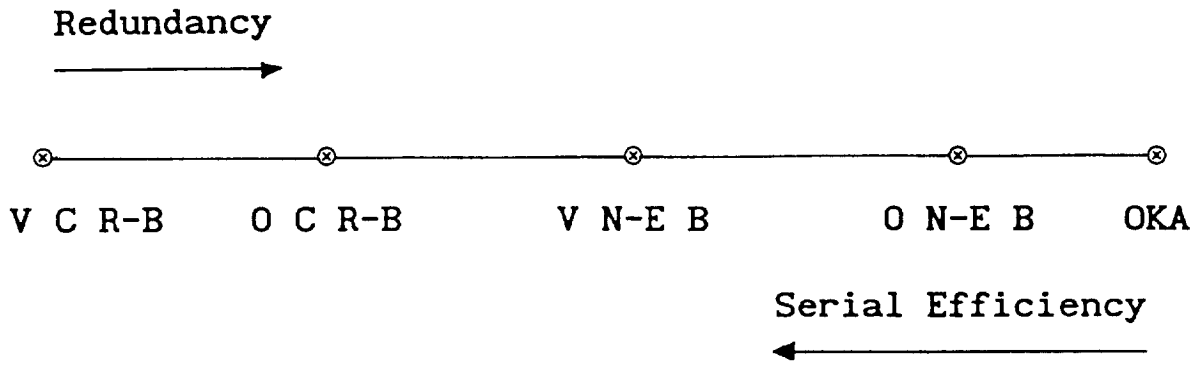
Comparison of Serial Efficiency of Different Algorithms

In order to study the relative efficiency of the algorithms, the optimal choice of coordinate frame(s) for projection of the Equations should be carefully analyzed.

For the Variant of Newton-Euler Algorithm, projection of all equations onto any fixed frame leads to maximum computational efficiency; It requires $O(n)$ transformations.
Projection onto the body frame leads to copmputational inefficiency; it requires $O(n^2)$ transformations!

For the Variant of Composite Rigid-Body Algorithm, projection of Step 1 onto body frame and Step 2 onto any fixed frame leads to maximum computational efficiency; It requires $O(n)$ transformations.
Projection of both steps onto the body frame leads to copmputational inefficiency; it requires $O(n^2)$ transformations!

Comparison of Serial Efficiency of Different Algorithms



OKEA: Original Kane's Equation Algorithm.

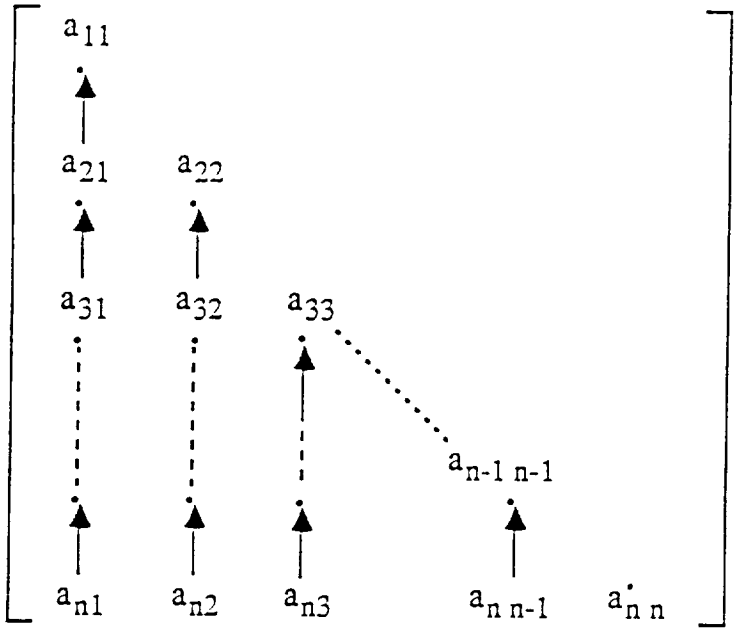
O N-E B: Original Newton-Euler Based Algorithm.

V N-E B: Variant of Newton-Euler Based Algorithm.

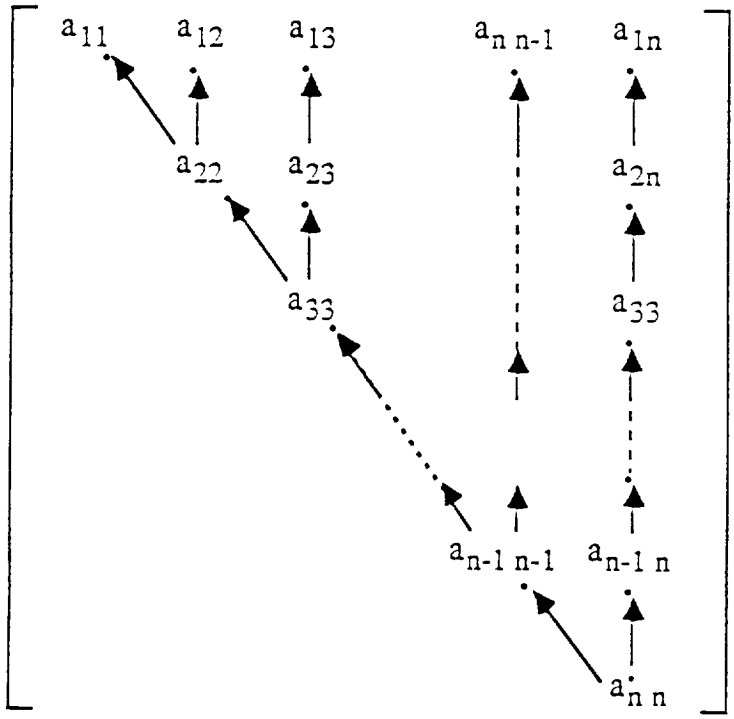
O C R-B: Original Composite Rigid-Body Algorithm.

V C R-B: Variant of Composite Rigid-Body Algorithm.

Algorithm	General		n = 6		
	Mul.	Add.	Mul.	Add.	Total
V N-E B	$(39/2)n^2+$ $(195/2)n-95$	$19n^2+$ $55n-66$	1192	948	2140
V C R-B	$(9/2)n^2+$ $(231/2)n-181$	$4n^2+$ $88n-137$	644	535	1179



(a)



(b)

Computational Structure of and Data-Dependency in Algorithms for Mass Matrix
a) The Newton-Euler Based Algorithms
b) The Composite Rigid-Body Algorithms

Algorithmic Choice for Parallel Computation of Mass Matrix

Parallelism in Computation of Mass Matrix: Time and Processors Bouds

We have shown that the time lower bound in computation of mass matrix is of $O(\log_2 n)$ and can be achieved by using $O(n^2)$ processors.

The Original Kane's Equation Algorithm might seem very suitable for parallel computation since all elements of the mass matrix can be computed totally in parallel.
The computation of each element of mass matrix can be performed in $O(\log_2 n)$ steps by using $O(n)$ processors. Hence, in order compute all the elements in parallel and achieve the time lower bound of $O(\log_2 n)$, $O(n^3)$ processors are required!

Using both the Newton-Euler Based Algorithms and the Composite Rigid-Body Algorithms, the mass matrix can be computed in $O(\log_2 n)$ steps with only $O(n^2)$ processors.

Algorithmic Choice for Parallel Computation of Mass Matrix

The Newton-Euler Based Algorithms are more suitable for parallel computation due to their regular computational structure and a lesser degree of data-dependency in their computation.

1) They provide a high degree of coarse grain parallelism:
The columns of the mass matrix can be computed in parallel.

2) They are more regular and have a finer grain:
A higher degree of parallelism in computation of the elements of each column can be exploited

3) Their parallel computation on a two-dimensional processor array requires simpler communication and synchronization mechanisms.

Choice of Coordinate Frame for Parallel Computation on a two-dimensional processor array:

For the Variant of Newton-Euler Based Algorithm it is more efficient to project the equations of onto the End-effector (Body n) frame while for the Variant of Composite Rigid-Body Algorithm it is more efficient to project the equations onto the base frame!

Conclusion

* For recursive serial computation, the Variant of Composite Rigid-Body Algorithm is significantly more efficient than the Variant of Newton-Euler and the Variant of Kane's Equation Algorithms.

* For parallel computation with $O(n^2)$ processors, i.e., maximum exploitation of parallelism, the Variant of Newton-Euler and the Variant of Kane's Equation Algorithms are not only significantly more efficient than the Variant of Composite Rigid-Body Algorithm but they also require much simpler architectural features.

* For parallel computation with $O(n)$ processors, i.e., limited exploitation of parallelism, the Variant of Composite Rigid-Body Algorithm is more efficient than the Variant of Newton-Euler and the Variant of Kane's Equation Algorithms

Comparison of Two Classes of Serial and Parallel Algorithms for Computation of Mass Matrix

Algorithm		Computation Cost		SP	Proc.
		General	n = 6		
SA	VCR-B	$((9/2)m+4a))n^2+((231/2)m+88a))n-(181m+137a)$	$644m+535a$	-	1
	VN-EB	$((39/2)m+19a)n^2+((195/2)m+55a)n-(95m+66a)$	$1192m+948a$	-	1
PA	VCR-B	$(48m+63a)\lceil\log_2 n\rceil+(100m+65a)$	$244m+254a$	2.40	n(n+1)/2
	VN-EB	$(33m+33a)\lceil\log_2 n\rceil+(109m+89a)$	$208m+188a$	2.98	n(n+1)/2
PPA	VCR-B	$(9m+8a)n+(48m+63a)\lceil\log_2 n\rceil+(58m+24a)$	$256m+261a$	2.32	n
	VN-EB	$(39m+38a)n+(27m+18a)\lceil\log_2 n\rceil+(25m-2a)$	$340m+280a$	1.90	n

SA: Serial Algorithm.

PA: Parallel Algorithm with $O(n^2)$ processors.

PPA: Parallel Algorithm with $O(n)$ processors.

Parallel VNEB algorithm

Step 1:

1) Parallel compute R(j+1, j) by all processors of Row j.

For j = 1, 2 ,..., n

  For i = 1, 2, ..., j

    PR_{ji} : R(j+1, j)

2) Parallel compute R(n+1, j) by processors of Column i.

For i = 1, 2, ..., n

  For j = i, i+1, ..., n

    For η = 1 step 1 until $\lceil \log_2(n+1-i) \rceil$, Do

      $R(j+2^{\eta}, j) = R(n+1, j)$

      $j+2^{\eta} > j+2^{\eta-1} \geq n+1$

      $R(j+2^{\eta}, j) = R(n+1, j) = R(n+1, j+2^{\eta-1})R(j+2^{\eta-1}, j)$

      $j+2^{\eta} \geq n+1 > j+2^{\eta-1}$

      $R(j+2^{\eta}, j) = R(j+2^{\eta}, j+2^{\eta-1})R(j+2^{\eta-1}, j)$

      $n+1 > j+2^{\eta} > j+2^{\eta-1}$

    End_Do

3) Shift R(n+1, j+1) by processors of Row j+1 to the processors of Row j.

For j = 1, 2, ..., n

  For i = 1, 2, ..., j

    PR_{ji}: R(n+1, j+1)

    with R(n+1, n+1) = U (Unit Matrix)

4) Parallel compute ${}^{n+1}Z(j)$, ${}^{n+1}P(j+1,j)$, and ${}^{n+1}H(j)$ by all processors of Row j.

For $j = 1, 2, \ldots, n$

For $i = 1, 2, \ldots, j$

a) PR_{ji} : ${}^{n+1}Z(j) = R(n+1,j)\,{}^{j}Z(j)$

with ${}^{j}Z(j) = [0\ 0\ 1]^{t}$

b) PR_{ji}: ${}^{n+1}P(j+1,j) = R(n+1,j+1)\,{}^{j+1}P(j+1,j)$

c) PR_{ji}: ${}^{n+1}S(j) = R(n+1,j+1)\,{}^{j+1}S(j)$

d) PR_{ji}: ${}^{n+1}H(j) = M(j)\,{}^{n+1}S(j)$

Step 2:

1) Parallel compute $P(j+1,i)$ and $\omega(j,i)$ by processors of Column i.

For $i = 1, 2, \ldots, n$

For $j = i, i+1, \ldots, n$

For $\eta = 1$ step 1 until $\lceil \log_2(n+1-i) \rceil$, Do

$\omega(j+2^{\eta},i) = \omega(j+2^{\eta-1},i) = Z(i)$

$P(j+2^{\eta},j) = P(j+1,i)$

$j+2^{\eta} > j+2^{\eta-1} \geq n+1$

$P(j+2^{\eta},j) = P(j+1,i) = P(j+2^{\eta},j+2^{\eta-1}) + P(j+2^{\eta-1},j)$

$j+2^{\eta} \geq n+1 > j+2^{\eta-1}$

$P(j+2^{\eta},j) = P(j+2^{\eta},j+2^{\eta-1}) + P(j+2^{\eta-1},j)$

$n+1 > j+2^{\eta} > j+2^{\eta-1}$

End_Do

2) Parallel compute V(j,i), F(j+1,j,i), and N(j+1,j,i) by

processors of Column i.

For i = 1, 2, ..., n

For j = i, i+1, ...n

a) PR_{ji}: $V(j,i) = \omega(j,i) \times P(j+1,i) = Z(i) \times P(j+1,i)$

b) PR_{ji}: $F(j+1,j,i) = \omega(j,i) \times H(j) + M(j) V(j,i)$

c) PR_{ji}: $N(j+1,j,i) = R(n+1,j+1)\left[{}^{j+1}K(j)R(j+1,n+1)^{n+1}\omega(j,i)\right] +$

$H(j) \times V(j,i)$

Step 3:

1) Parallel compute F(n+1,j,i) processors of Column i.

For i = 1, 2, ..., n

For j = i, i+1, ..., n

For η = 1 step 1 until $\lceil \log_2(n+1-j) \rceil$, Do

$F(j+2^{\eta}, j, i) = F(n+1, j, i)$

$j+2^{\eta} > j+2^{\eta-1} \geq n+1$

$F(j+2^{\eta}, j, i) = F(n+1, j, i) = F(j+2^{\eta}, j+2^{\eta-1}, i) + F(j+2^{\eta-1}, j, i)$

$j+2^{\eta} > n+1 > j+2^{\eta-1}$

$F(j+2^{\eta}, j, i) = F(j+2^{\eta}, j+2^{\eta-1}, i) + F(j+2^{\eta-1}, j, i)$

$n+1 > j+2^{\eta} > j+2^{\eta-1}$

End_Do

2) Shift $F(n+1, j+1, i)$ by processors of Row $j+1$ to processors of Row j.

For $j = 1, 2, \ldots, n$

For $i = 1, 2, \ldots, j$

PR_{ji}: $F(n+1, j+1, i)$

3) Parallel compute $N(n+1, j, i)$ by processors of Column i.

For $i = 1, 2, \ldots, n$

For $j = i, i+1, \ldots, n$

a) PR_{ji}: $N(j+1, j, i) = N(j+1, j, i) + P(j+1, j) \times F(n+1, j+1, i)$

b) For $\eta = 1$ step 1 until $\lceil \log_2(n+1-j) \rceil$, Do

$$N(j+2^{\eta}, j, i) = N(n+1, j, i) \qquad j+2^{\eta} > j+2^{\eta-1} \geq n+1$$

$$N(j+2^{\eta}, j, i) = N(n+1, j, i) = N(n+1, j+2^{\eta-1}, i) + N(j+2^{\eta-1}, j, i) \qquad j+2^{\eta} \geq n+1 > j+2^{\eta-1}$$

$$N(j+2^{\eta}, j, i) = N(j+2^{\eta}, j+2^{\eta-1}, i) + N(j+2^{\eta-1}, j, i) \qquad n+1 > j+2^{\eta} > j+2^{\eta-1}$$

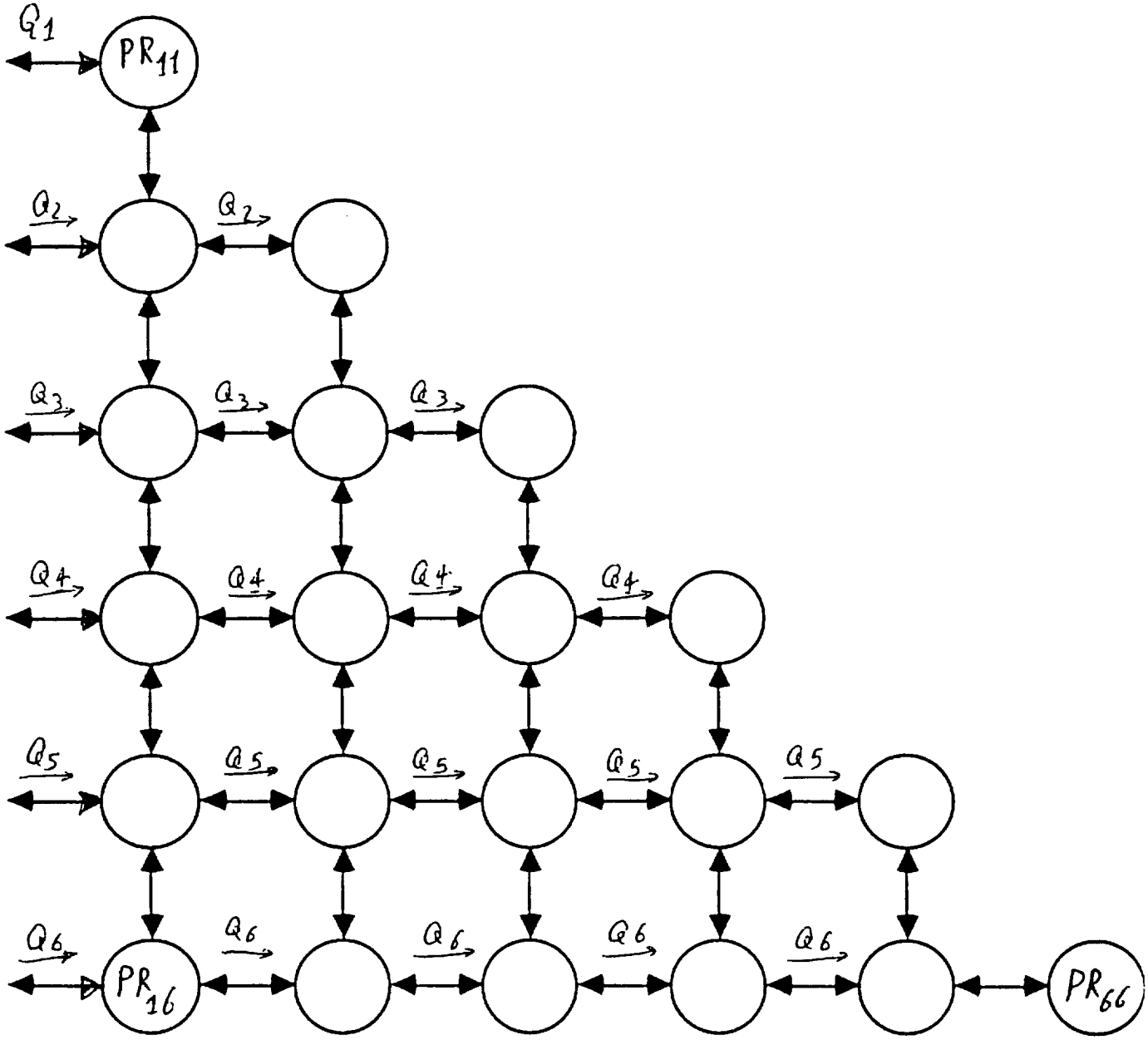
End_Do

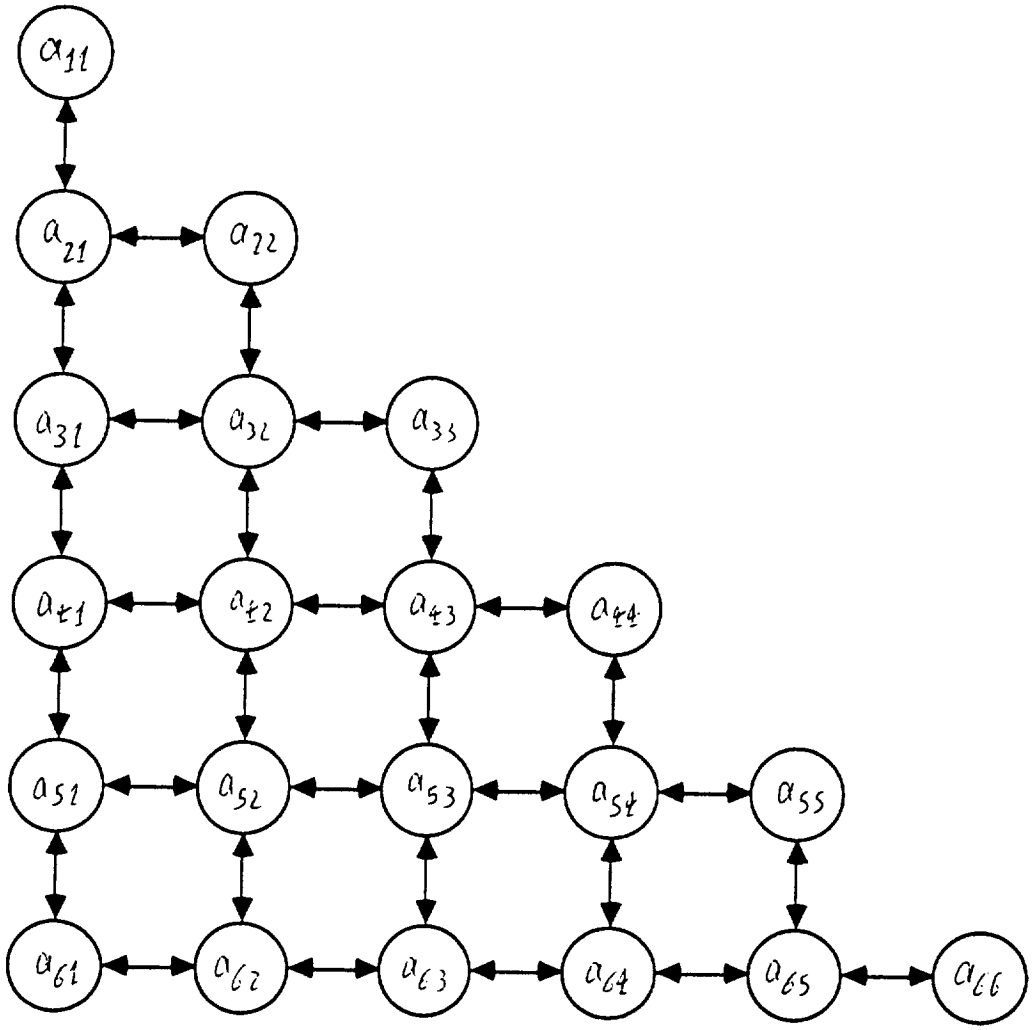
2) Parallel compute a_{ji} by PR_{ji}.

For $i = 1, 2, \ldots, n$

For $j = i, i+1, \ldots n$

PR_{ji}: $a_{ji} = Z(j).N(n+1, j, i)$


Q1
PR11
Q2
Q2
Q3
Q3
Q3
Q4
Q4
Q4
Q4
Q5
Q5
Q5
Q5
Q5
Q6
Q6
Q6
Q6
Q6
PR16
PR66


a_{11}
a_{21}
a_{22}
a_{31}
a_{32}
a_{33}
a_{41}
a_{42}
a_{43}
a_{44}
a_{51}
a_{52}
a_{53}
a_{54}
a_{55}
a_{61}
a_{62}
a_{63}
a_{64}
a_{65}
a_{66}

Algorithm-To-Architecture Mapping

Determination of an Algorithmically-Speciaslized Parallel Architecture for Efficient Implementation of the Algorithm.

1) Processors Interconnection and Communication Complexity

For perfect mapping:

a) The required interconnection among processors of each column is Shuffle Exchange augmented with Nearest-Neighbor (SENN).

b) The required interconnection among processors of each row is Nearest-Neighbor.

The perfect mapping leads to a communication complexity of $O(\log_2 n)$. Mapping on an array with nearest-neighbor interconnection leads to the communication complexity of $O(n)$.

2) Synchronization Mechanism

Exploitation of parallelism at two computational levels:

a) Coarse grain parallelism in computing columns of mass matrix, and

b) Fine grain parallelism in computing the elements of each column.

Global Clock-Based Synchronization Mechanism (similar to Systolic Array) for processors of each column, and Local Data Driven (similar to Wavefront Array) for processor of each row.

7533

A GENERIC MULTI-FLEX-BODY DYNAMICS, CONTROLS SIMULATION TOOL FOR SPACE STATION

N91-22321

Ken W. London*, PhD and John F. L. Lee**, PhD
Honeywell, Inc., Clearwater, Florida

Ramen P. Singh***, PhD and Buddy Schubele****
DYNACS Engineering, Inc., Clearwater, Florida

ABSTRACT

An order (n) multi-flex body Space Station simulation tool is introduced. The flex multibody modeling is generic enough to model all phases of Space Station from build up through to Assembly Complete configuration and beyond. Multibody subsystems such as the Mobile Servicing System (MSS) undergoing a prescribed translation and rotation are also allowed. The software includes aerodynamic, gravity gradient and magnetic field models. User-defined controllers can be discrete or continuous. Extensive preprocessing of body-by-body NASTRAN flex data is built in. A significant aspect, too, is the integrated controls design capability which includes model reduction and analytic linearization.

1.0 INTRODUCTION

The buildup of the International Space Station Freedom evolves through a series of widely differing configuration each with its own unique mass property distribution and ensuing stability and controls issues. Even the relatively mature Assembly Complete configuration can itself undergo significant variation in mass properties when the MSS is in transit and/or carrying out payload maneuvers or during Orbiter docking. In addition to the complications introduced by the multibody articulated nature of these configurations, there is the fact that each component assembly is not rigid but possesses a degree of structural flexibility. To support the controls design and verification for such complex orbiting dynamic systems requires a versatile and high fidelity simulation capability.

This paper describes a dynamics, controls time history simulation software package (SSSIM Rev. 2.0)‡ that specializes a very generic multibody topology to the needs of Space Station with flexibility effects included and is a significant extension of the rigid model described in Ref. 1. Flex modeling is based on a finite element 'nodal' representation compatible with NASTRAN. An efficient preprocessor is used to extract the necessary mass, stiffness and damping characteristics as well as the dynamic rigid/flex coupling coefficients. Calculations can be simplified considerably depending on the nature of the boundary conditions. To minimize the system order, a model reduction feature is built into the code. An interactive setup program aids the user in constructing the data file.

---

* Staff Engineer
** Senior Engineering Fellow
*** President
**** Aerospace Engineer

‡ Developed under subcontract to Honeywell (SSG), Inc. by DYNACS Engineering, Inc.

The intent overall is to have a complete controls design analysis capability, hence, the code includes a series of idealized sensor options and a range of actuator types. The design process is assisted too by a newly developed analytic linearization capability which can linearize about nontrivial states and yields output in a form directly compatible with standard linear analysis tools such as MATLAB. Such features, when combined with a user-defined controller (continuous or discrete), provide a full closed loop large angle transient dynamic analysis tool for Space Station.

An overview follows of basic features and capabilities embodied in the software. As well, an outline is given of the approach used for the flex dynamics formulation and for the solution algorithm (numerous steps are taken to minimize simulation execute times). Preliminary results and run time performance are given for representative Space Station configurations.

2.0 CONFIGURATION AND FLEX DYNAMIC ANALYSIS

2.1 TOPOLOGY

The generic "tree" configuration topology forming the basis for the code is shown in Fig. 1. Any body that has more than one outboard body attached to it is labelled as a "base" body. A "leaf" body has no bodies outboard of it. A "branch" is made up of a chain of adjacent bodies starting with a base body innermost and ending with another base body or a leaf body. The "level" of a branch is a measure of how many base bodies there are between the branch base body and the reference (level 1) base body. Each body is defined on a stand-alone basis without regard to the rest of the bodies in the tree. The sensors, actuators and joints are connected to the bodies at specific locations called node points. Each individual body may be rigid or flexible.

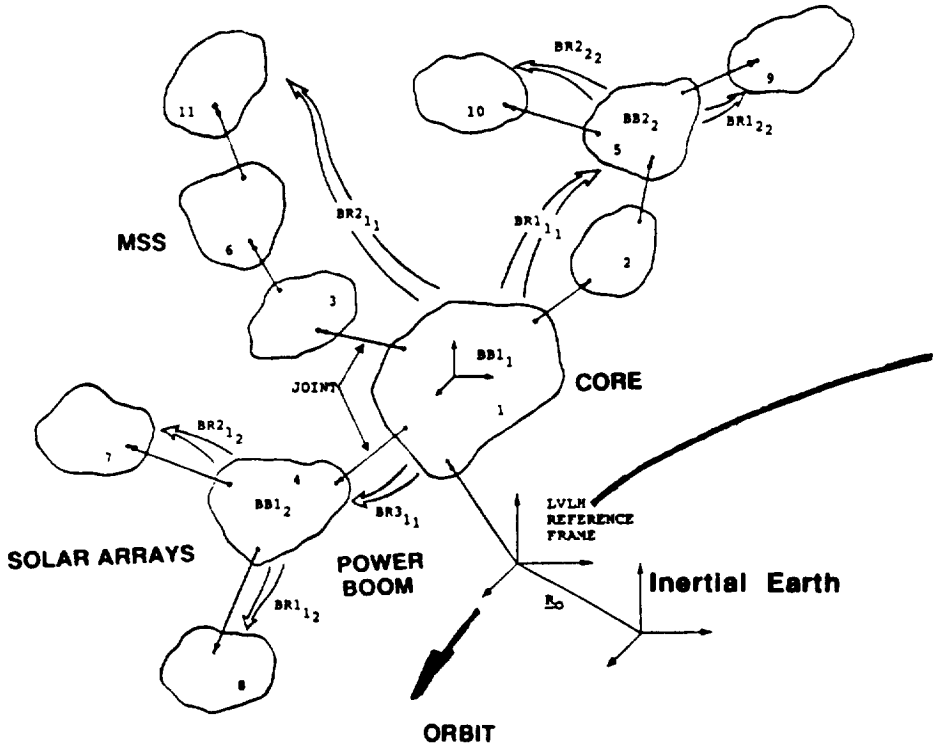


Fig. 1 Generic multibody configuration with a multiple branch, multiple base body hierarchy, point-connected interbody joints and a prescribed orbiting LVLH reference applied to Space Station.

For applications to Space Station, the structure is assumed to have a core body undergoing up to 3 translational and up to 3 rotational (attitude) degrees of freedom (DOF) relative to the local vertical, local horizontal (LVLH). The frame fixed to LVLH is, in turn, following some predetermined orbital trajectory. All other bodies can have up to one rotational degree of freedom or can have prescribed motions that can occur along one axis for rotation and along up to 3 axes in translation. As indicated in Fig. 1, for example, base body 1 (BB1) can represent the core body and base body 2 (BB2) can be a power boom with leaf bodies 7 and 8 being pairs of solar arrays.

The MSS itself is a complex multibodied subassembly which, when operational, can significantly alter the overall mass properties of the Space Station. MSS-related activies can also generate significant disturbance loads.Consequently, it has served as somewhat of a focal point in development of the SSSIM software. One objective is to have the code generic enough to capture the major dynamic impact of the MSS during its many different configurations and missions. The current software can represent the MSS as an assembly of any number of point-connected bodies. There is no restriction as to body location or as to individual body mass properties. Prescribed motions are possible at joints both in rotation and in translation and are general enough to allow plane change maneuvering. The base body must be rigid if prescribed motions are involved since there are no suitable NASTRAN flex modes available for a translating structure. Payload release is also readily accomodated.

2.2 DYNAMICS FORMULATION OVERVIEW

The order (n) dynamics developed in Ref. 2 for rigid multibody cases is extended to include elastic deformations. Now accelerations associated with the outboard body vibrations are first shifted inboard along with outboard body acceleration. It is beyond the scope of this presentation to go into full detail (Refs. 3, 4) but the basic impact of flexibility on displacement field is shown here in order to appreciate the fundamental nature of the model.

Figure 2 provides a description of body pairs *j* and *L(j)*. As seen in Figure 2, a "nodal" body representation for flexible bodies is assumed. The nodal body, b_{jo}, will facilitate representation of

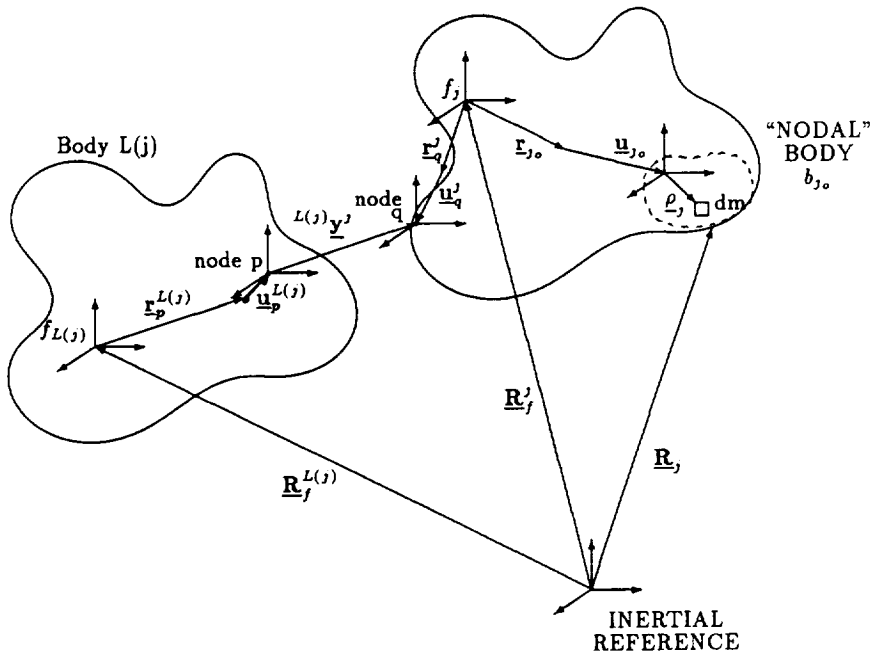


Figure 2 Inboard *(L(j))* outboard *(j)* 2-body geometry forms the basis for the recursive kinematics.

discrete nodal data from finite element program NASTRAN, and provides means of accounting for effects such as mass center offset and eccentricity in the dynamics. These effects can arise when, for example, lumped masses are attached at nodes when running NASTRAN. For a consistent finite element formulation, effects such as these result in a coupled nodal mass matrix. In addition to the effects such as eccentricity, it is worth pointing out that this nodal body concept also introduces an additional rotation for mass elements as evident in Figure 2 where the nodal body, b_{jo}, has its' own reference frame. Throughout this text, it will be assumed that the nodal body reference frames are aligned with the body reference frame when the generalized modal coordinates are zero. It is also assumed that the deformations are small so that linear, time invariant, modal data is used.

With the multibody description provided, it is evident that required body data (mass properties, geometry, modal data, etc.) is defined for each body separately in its own reference. The vector location of the mass element, *dm*, of nodal body b_{jo} of body j is in inertial as follows:

$$\underline{R}_j = \underline{R}_f^j + \underline{r}_{jo} + \underline{u}_{jo} + \underline{\rho}_j$$

where:

$\underline{R}_f^j$ ≡ Vector locating body j reference in inertial space

$\underline{r}_{jo}$ ≡ Undeformed location vector of nodal body b_{jo} reference wrt body j reference

$\underline{u}_{jo}$ ≡ Vector of linear deformation associated with nodal body b_{jo} reference wrt body j reference

$$\underline{u}_{jo} \triangleq \sum_{i=1}^{NMj} \underline{\phi}_i^j (\underline{r}_{jo})\, \eta_i^j$$

$\underline{\rho}_j$ ≡ Vector locating dm of nodal body b_{jo} wrt nodal body b_{jo} reference (fixed in nodal body b_{jo} basis)

The location vector of body j reference in inertial space is defined in a recursive sense via the location of its inboard body and relative joint displacements.

$$\underline{R}_f^j = \underline{R}_f^{L(j)} + (\underline{r}_p^{L(j)} + \underline{u}_p^{L(j)}) + {}^{L(j)}\underline{\nu}^j - (\underline{r}_q^j + \underline{u}_q^j)$$

where:

$\underline{R}_f^{L(j)}$ ≡ Vector locating body $L(j)$ (body directly inboard of body j) reference in inertial space

$\underline{r}_p^{L(j)}$ ≡ Vector locating undeformed reference point p (point fixed in Body L(j) for defining joint j) wrt body L(j) reference, fixed in body $L(j)$

$\underline{u}_p^{L(j)}$ ≡ Vector of linear deformation associated with point p wrt body L(j) reference, fixed in body $L(j)$

$$\underline{u}_p^{L(j)} \triangleq \sum_{i=1}^{NML(j)} \underline{\phi}_i^{L(j)}(\underline{r}_p^{L(j)})\, \eta_i^{L(j)}$$

${}^{L(j)}\underline{y}^j$ ≡ Vector of linear joint j displacement of reference point q in body j wrt reference point p in body $L(j)$, fixed in reference p of body $L(j)$

$${}^{L(j)}\underline{y}^j \triangleq \sum_{i=1}^{NTj} \underline{g}_i^j\, y_i^j$$

$\underline{r}_q^j$ ≡ Vector locating undeformed reference point q (point fixed in body j for defining joint j) wrt body j reference, fixed in body j

$\underline{u}_q^j$ ≡ Vector of linear deformation associated with reference point q wrt body j reference, fixed in body j

$$\underline{u}_q^i \triangleq \sum_{i=1}^{NMj} \underline{\phi}_i^j(\underline{r}_q^i)\, \eta_i^j$$

The coefficients of h, $\underline{\Phi}$, are referred to as "Mode Shape" vectors and are provided via NASTRAN. The model shape vectors are assumed to be time invariant consistent with the assumption that deformation, u, is small. The second generalized coordinate was that associated with joint j displacement, y. The relative displacement of point q in body j with respect to (wrt) point p in body $L(j)$ is written in terms of the orthogonal unit vectors, $\underline{g}^j$ (fixed in the material frame at point p in body $L(j)$) and the generalized translation coordinate y, where i is from 1 to the number of translational degrees of freedom of the joint, NTj.

NM_j is the number of modes of body j, NT_j is the number of translational degrees of freedom for joint j, and NR_j is the number of rotational degrees of freedom for joint j.

Velocity and acceleration follow by differentiation. Here though, only velocity is presented. The acceleration is even more complex. Local material frames will have a relative, local angular velocity with respect to body reference frames due to rotational deformation. For the most part, local rotation due to deformation at the nodal bodies will be neglected in as much as the associated coordinate transformations are not introduced (which would give rise to higher order terms). However, the relative orientation of reference frames particular points will include rotation due to deformation, for example, in computing body j to body $L(j)$ coordinate transformation. Like linear deformation and mode shape vectors, rotation deformation for local angular velocity will use the "Mode Slope Vectors", ϕ', also available from NASTRAN.

Inertial velocity of mass element, dm, of nodal body b_{jo} of body j follows. An open dot above a vector will be used to denote local velocity, whereas a solid dot above a vector will be used to denote an inertial velocity.

Performing the differentiation of the displacement field, the inertial velocity of the mass element is as follows.

$$\dot{\underline{R}}_j = \dot{\underline{R}}_f^j + \underline{\omega}^j \times (\underline{r}_{jo} + \underline{u}_{jo}) + \overset{\circ}{\underline{u}}_{jo} + \underline{\omega}_o^j \times \underline{\rho}_j$$

where:

$$\dot{\underline{R}}_f^j = \dot{\underline{R}}_f^{L(j)} + \underline{\omega}^{L(j)} \times (\underline{r}_p^{L(j)} + \underline{u}_p^{L(j)}) + \overset{\circ}{\underline{u}}_p^{L(j)} + \underline{\omega}_p^{L(j)} \times {}^{L(j)}\underline{v}^j$$

$$+ \quad {}^{L(j)}\overset{\circ}{\underline{v}}^j - \underline{\omega}^j \times (\underline{r}_q^j + \underline{u}_q^j) - \overset{\circ}{\underline{u}}_q^j$$

$\underline{\omega}^j \equiv$ Angular velocity vector of body j reference in inertial space

$$\underline{\omega}^j \triangleq \underline{\omega}^{L(j)} + \overset{\circ}{\underline{u}}_p'^{L(j)} + {}^{L(j)}\underline{\omega}^j - \overset{\circ}{\underline{u}}_q'^j$$

$\underline{\omega}^{L(j)} \equiv$ Angular velocity vector of body $L(j)$ reference in inertial space

$\underline{\omega}_p^{L(j)} \equiv$ Angular velocity vector of reference at point p in body $L(j)$ wrt inertial space

$$\underline{\omega}_p^{L(j)} \triangleq \underline{\omega}^{L(j)} + \overset{\circ}{\underline{u}}_p'^{L(j)}$$

$$\overset{\circ}{\underline{u}}_p'^{L(j)} \triangleq \sum_{i=1}^{NML(j)} \underline{\phi}_i'^{L(j)}(\underline{r}_p^{L(j)})\, \dot{\eta}_i^{L(j)}$$

$${}^{L(j)}\overset{\circ}{\underline{v}}^j \equiv \sum_{i=1}^{NTj} \underline{g}_i^j\, \dot{v}_i^j$$

${}^{L(j)}\underline{\omega}^j \equiv$ Angular velocity vector of reference q wrt reference p

$${}^{L(j)}\underline{\omega}^j \triangleq \sum_{i=1}^{NRj} \underline{e}_i^j(\theta^j)\, \dot{\theta}_i^j$$

$\underline{\omega}_o^j \equiv$ Angular velocity vector of nodal body b_{jo} reference in inertial space

$$\underline{\omega}_o^j \triangleq \underline{\omega}^j + \overset{\circ}{\underline{u}}_{jo}'$$

$$\overset{\circ}{\underline{u}}_{jo}' \triangleq \sum_{i=1}^{NMj} \underline{\phi}_i'^j(\underline{r}_{jo})\, \dot{\eta}_i^j$$

Also,

$$\overset{\circ}{\underline{u}}_q'^j \triangleq \sum_{i=1}^{NMj} \underline{\phi}_i'^j(\underline{r}_q^j)\, \dot{\eta}_i^j$$

Clearly, flexibility introduces considerable additional algebraic detail, augmenting not just the displacement field but contributing significantly to both translation and rotational velocity.

The Newton, Euler dynamic equation can now be formulated for a 2 body point-connected pair and solved for recursively. The kinematics, dynamics is transformed successively body by body from outermost to the most inboard body frame (which need not be situated at the mass center). Accelerations are then determined for this innermost body based on augmented inertia properties. This solution is propagated outboard to solve for the accelerations of all other bodies. The net effect is a solution algorithm that avoids the need for direct numerical inversion of a high order system mass matrix. Consequently, run time is proportional to number of degrees of freedom (n) - i.e. order (n) - and not to n^2 or n^3 say [2].

3.0 OVERVIEW OF CAPABILITY

3.1 BASIC SUPPORT FEATURES

In addition to high fidelity modeling of the multibody dynamics, the SSSIM Rev. 2.0 software provides the same basic support features common in Ref. 1. These include detailed models of the primary environmental loads in orbit. i.e. the Earth's gravitational effect (including oblateness) and low density aerodynamics (1970 Jacchia atmosphere) applied to flat plate or cylindric surfaces. A magnetic field model is available too and is in use for passive damper studies for early configurations. For controls design purposes, the code includes a series of idealized options such as position sensors, single-axis rate gyros and accelerometers, resolvers, tachometers as well as a control moment gyro (CMG) resolver. Similarly, a range of actuator types are provided which include: reaction jets, torque motors, torque devices, magnetic torques and double-gimbal CMG's. User defined controllers can be continuous or discrete.

The basic simulation capability described above is complemented by a series of enhancements that model disturbance associated with:

- solar radiation pressure
- CMG, torque motor friction effects
- fuel slosh (3 DOF point mass and nonlinear stiffness, damping)
- static, dynamic imbalance for rotating bodies
- fans
- fluid flow (pumps, heat exchangers, throttles)
- start/stop friction of solar array drives, antenna pointing mechanisms
- crew actions (kick-off, sneezing, walking).

Such detailed modeling makes it possible to assess impact of such disturbances on acceleration level at any specified location (i.e. micro "g" impact).

3.2 SUMMARY OF MODELING CAPABILITIES AND LIMITATIONS

Structure. The structure is composed of bodies and joints connected in an open tree topology. You can have any number of bodies. There are two limitations on topology. The joints between adjacent bodies are pin connected and no closed loops ar are allowed.

Bodies. The bodies in SSSIM may be rigid or flexible. All of the data for each individual body is defined in that body's reference frame. For flexible bodies the boundary conditions are arbitrary, i.e., fixed-free, etc. The finite element models for flexible bodies can have point masses or rigid bodies called nodal bodies at each one of the node points. There are 3 limitations on the definitions of bodies. The first is that only normal modes may be used, the second is that deformations must be small, and the third is that the third degree terms in deformation are neglected.

Joints. Each joint connects two adjacent bodies. The degrees of freedom at each joint can be prescribed, free, or locked and large angle rotations and translations are allowed. Initial values of the coordinates for locked degrees of freedom are allowed. Joint 1 connects the core body to the local vertical reference frame and is treated differently from all the remaining joints. The number of prescribed and free degrees of freedom is limited as defined in Table 1. An additional limitation is that for each joint with prescribed translation, the inboard body must be rigid.

TABLE 1 JOINT DOF

	Allowable Number of DOF's	
	Joint 1	Joints 2-N
Free Rotation	0,1,2,3	0,1
Prescribed Rotation	0	0,1[a]
Free Translation	0,3	0
Prescribed Translation	0	0,1,2,3

[a]If the i^{th} joint has 1 free rotation DOF then it must have 0 prescribed rotation DOF's

Sensors and Actuators. Sensors and actuators are generic in their types and placement. Your can select the types of sensors and actuators from a menu and place them at any node in the structure. The sensors and actuators are limited in that they do not include dynamics or noise with the exception of CMG's and torque motors. CMG dynamics are included and fully coupled with the structure dynamics. Both the CMG and the torque motors have a friction model which may be considered a noise source.

Controllers. The controllers have unlimited capability because they are user supplied subroutines. The continuous controller is called inside the fourth order Runge-Kutta integration loop and the discrete controller is called outside the loop at discrete instants in time. Controllers are limited to only one controller per simulation.

Orbiter Environment. The orbit environment includes a standard NASA atmosphere and magnetic field model. The atmosphere model includes the diurnal bulge and the effect of solar flux and magnetic activity. The primary limitation on the orbit environment is that the orbit is fixed and defined a priori. This means that any forces and moments acting on the structure due to reaction jets, aerodynamic drag, etc. will not change the orbit. The orbital equations are uncoupled form the dynamic equations and solved assuming zero disturbance acceleration. The effect of gravity gradient on the multi-body structure is modeled as a point force and moment applied at the reference frame of each individual body, where the distributed effect of the gravitational disturbance on flexibility has been neglected.

3.3 SPECIAL FEATURES AND ADDED CAPABILITY

The basic SSSIM software has been enhanced considerably in order to support the flex modeling; in order to create a controls analysis tool and in order to improve user friendliness and efficiency. This section outlines the major areas of enhancement.

SETUP Program

A considerable volume of data can be required to run SSSIM Rev. 2 as the number of bodies and joints grow and with full-up implementation of controllers. A single error in the data file can stop a run and it can be a painstaking process to track down. Program SETUP is designed to assist the user interactively in building up an error-free data file. Data is entered block by block, e.g. body 1, body 2, ...; joint 1, joint 2, ...etc. The user can add, delete and modify any given block of data and error checking is provided.

PREFLX Program

The flex model, as shown earlier, is based on a linear assumed mode representation of the elastic displacement. During solution, the deformation is separated into a space dependent part and a time dependent part. Furthermore, the spatial modal component is discretized into displacements for a series of interconnected 'nodal' mass elements analogous to the finite element models used in NASTRAN. This type of modeling ensures compatibility with contractor-supported structural data.

The purpose of the PREFLX program is to interface with NASTRAN data to form all the modal coefficients and terms needed by SSSIM Rev. 2 to accommodate flex degrees of freedom and to ensure rigid, elastic coupling.

Standard terms include the zeroth order terms $\underline{\alpha}_i^j$, $\underline{h}_i^j$ and the rigid body inertia dyadic, $\mathbb{I}_{rigid}^j$. The optional terms include the first and second order terms $\mathbb{M}_i^j$, $\underline{Y}_{ki}^j$ and $\mathbb{P}_{ki}^j$.

Quantities computed by PREFLX are expressed in the nodal body formulation as:

Zeroth Order Terms

$$\underline{\alpha}_i^j = \frac{1}{m_j}\left\{\sum_{0=1}^{NNB_j}\int_{b_{jo}} \underline{\phi}_{o_i}^j + \underline{\phi}_{o_i}^{\prime j} \times \underline{\rho}_j dm\right\}$$

$$\underline{h}_i^j = \sum_{o=1}^{NNB_j}\int_{b_{jo}} (\underline{r}_{jo} + \underline{\rho}_j) \times \underline{\phi}_{o_i}^j + (\underline{r}_{jo} + \underline{\rho}_j) \times (\underline{\phi}_{o_i}^{\prime j} \times \underline{\rho}_j) dm$$

$$\mathbb{I}_{RIGID}^j = \sum_{0=1}^{NNB_j}\int_{b_{jo}} \left((\underline{r}_{jo} + \underline{\rho}_j) \cdot (\underline{r}_{jo} + \underline{\rho}_j)\right)\mathbb{U} - (\underline{r}_{jo} + \underline{\rho}_j) \cdot (\underline{r}_{jo} + \underline{\rho}_j) dm$$

First Order Terms

$$\mathbb{M}_i^j = \sum_{0=1}^{NNB_j}\int_{b_{jo}} (\underline{r}_{jo} + \underline{\rho}_j) \cdot \underline{\phi}_{o_i}^j)\mathbb{U} - (\underline{r}_{jo} + \underline{\rho}_j)\underline{\phi}_{o_i}^j dm$$

$$\mathbb{N}_i^j = \mathbb{M}_i^{jT}$$

$$\underline{Y}_{ki}^j = \sum_{0=1}^{NNB_j}\int_{b_{jo}} \underline{\phi}_{o_k}^j \times \underline{\phi}_{o_i}^j + \underline{\phi}_{o_k}^j \times (\underline{\phi}_{o_i}^{\prime j} \times \underline{\rho}_j))dm$$

Second Order Term

$$\mathbb{P}_{ki}^j = \sum_{0=1}^{NNB_j}\int_{b_{jo}} (\underline{\phi}_{o_k}^j \cdot \underline{\phi}_{o_i}^j)\mathbb{U} - \underline{\phi}_{o_k}^j \underline{\phi}_{o_i}^j dm$$

where,

m_j = Mass of j^{th} Body

$\underline{\rho}_j$ = Location of dm wrt nodal body refrence frame

$\underline{r}_{jo}$ = Undeformed location of nodal body refrence frame

$\underline{\phi}^j_{oi}$ = Mode Shape of i^{th} Mode at o^{th} Node on j^{th} body

$\underline{\phi}'^j_{oi}$ = Mode Slope of i^{th} Mode of o^{th} Node on j^{th} body

$\{\cdot\}$ = 3 x 1column matrix

$\{\tilde{\cdot}\}$ = Skew symmetric matrix

nj = The number of NASTRAN structural nodes

nm = The number of retained flexible modes

r_c = Rigid body mass center location wrt the body refrence frame

$M_{nm \cdot nm}$ = Modal mass matrix

It is evident that considerable computational effort can be required. PREFLX is currently designed to accommodate 2500 node points and up to 250 modes per body.

LINEAR Program

Much control design is based on analysis of a linear systems model. An independent code LINEAR is in place which provide linear analytic modes (about zero or nonzero states). The topology is provided by SETUP. The analytic approach is considerably more accurate and more computationally efficient than its numerical counterpart. The output of this program is deliberately formatted for compatatibility with linear analysis tools such as MATLAB.

MODRED Program

MODRED performs model reduction on the linearized multi-flex-body output fro LINEAR at the component or system level. Three options are provided the user: (1) modified Component Cost Analysis (CCA); (2) CCA with special choice of parameters that yield costs based on controllability, observability Grammians; and (3) system model reduction using a p-q Covariance Equivalent Realization projection approach. Note that component model reduction preserves original coordinates allowing trace back to original flex body source data.

RESTART Option

Long duration runs are typical for Space Station, sometimes just to let the controller establish steady state. With this in mind, a RESTART option is provided to allow a run to be extended. At the end of a run the final system state is stored and used to initialized the data file for the next starting time.

3.4 COMPUTATIONAL CONSIDERATIONS

High fidelity dynamic simulation for an orbiting multibodied system such as Space Station can be computationally intensive. Run times can easily stretch into many hours even for the all-rigid cases. The problem becomes much more serious for flex modeling. Hence, the importance of order (n) modeling and solution. It is the flex cases that require the really high order models.

Other methods are needed to control run times too because of the higher flex frequencies which tend to dictate smaller integration step size. The number of modes retained can be kept to a minimum by carrying out model reduction. The software also provides 'options' to compute only up to zeroth, to first or to second order flex effects. A further innovation is the separate and independent coding of flex equations for differing boundary conditions.

The controls design is sped up too for flex cases by having an 'analytic' as opposed to a 'numeric' linearization. At all stages of development great care is exercised in storing and accessing the potentially large amount of flex data.

In addition to the flex-related steps taken above, the software also minimizes coordinate transformations by allowing only the core body to have up to 6 degrees of freedom whereas all other joints can have up to 1 rotation only (in addition to prescribed motions).

The RESTART capability does not speed up a run but it can remove the need to rerun time histories. Previous runs can be used to start up other longer term case studies as needed.

4.0 THE INTEGRATED SIMULATION, CONTROL SOFTWARE ENVIRONMENT

The capabilities and features discussed are integrated to provide a single time history simulation as well as a frequency domain controls analysis environment as shown in Fig. 3. The Space Station data can include multi-rigid-body configuration and mass properties and/or multi-flex-body NASTRAN structure models.

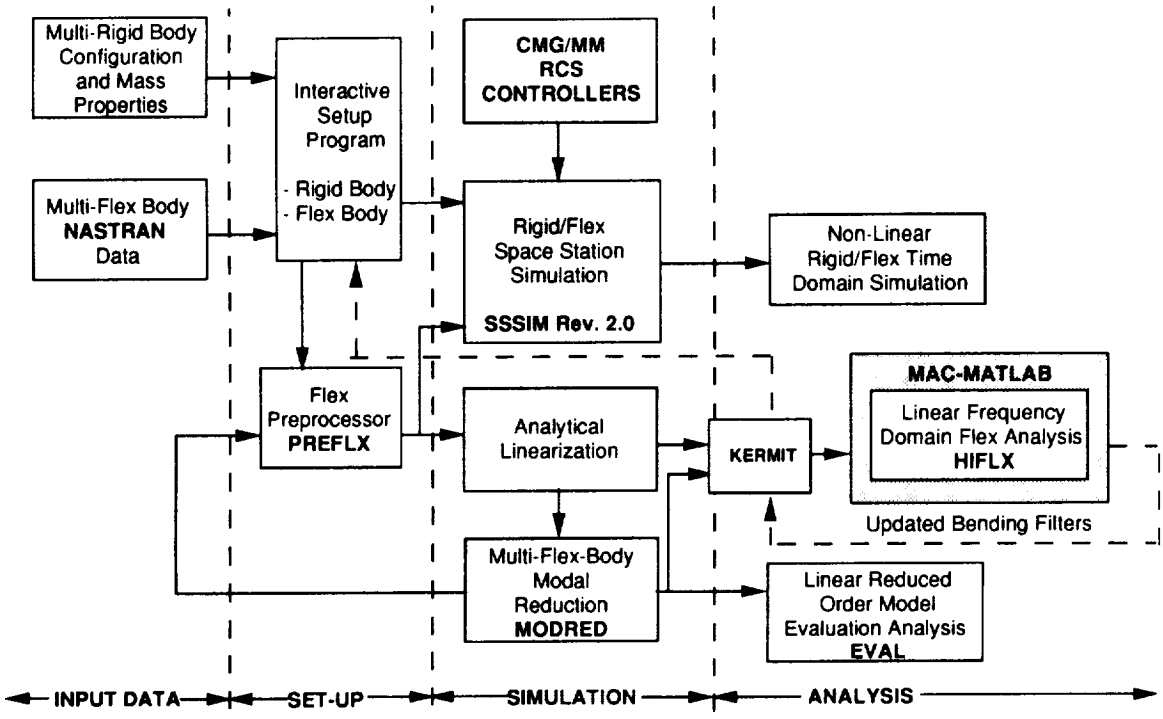


FIG. 3 MULTI-RIGID/FLEX-BODY DYNAMICS AND CONTROLS ANALYSIS/SIMULATION OVERVIEW

The separate stand-alone programs SETUP, PREFLX, SSSIM Rev. 2, LINEAR, MODRED, EVAL AND HIFLX that make up the SSSIM analysis environment can be applied in various combinations to perform 1. non-linear time history simulation, 2. model reduction, 3. linear system analysis, or 4. micro-gravity studies. The 6 programs are connected via data files that are created by one program and read by another without any user interaction. Three of these programs are interactive: SETUP is the basic model building program, MODRED and EVAL are used for model reduction and prompt the user for model reduction criteria. It is to be noted that all programs reside on a single VAX computer except for HIFLX, which is MacIntosh (MAC) based. Data transfer between the VAX and MAC is carried out via the KERMIT program which is a file transfer protocol.

Non-linear, Time-history Simulation. Time-history simulation is primary a three step process using SETUP, PREFLX AND SSSIM. The first step in this process is to define a model and enter the data using SETUP. The structure model is composed of bodies, joints, sensors and actuators.

The second step is to create the flex body data files. The bulk of the data for flexible bodies is handled automatically by PREFLX. A finite element model is created and used by NASTRAN to produce a modal model. PREFLX will read the modal model, compute the necessary modal integrals and form the flex data input file for the SSSIM Rev. 2 core. Once the number of nodes and modes are selected in SETUP, PREFLX takes over.

The third step is to simply execute the time-history simulation program, SSSIM.

Model Reduction. Model reduction is an iterative process. The first step is to create a full order model by running SETUP and PREFLX for the time-history simulation. Then instead of running SSSIM to do the simulation LINEAR is run to create a linear model of the system. The reduced order model is created by truncating flex modes from the full order model just created. MODRED and EVAL rank the modes and help select the retained bodes for the reduced order model. PREFLX is then re-run to create the flex data files for the reduced order model.

Linear System Analysis. A linear model of the system is created by running SETUP, PREFLX AND LINEAR to produce a linear model in the ABCD matrix quadruple format. HIFLX then can add the orbital dynamics, rigid body state estimator dynamics and Control Moment Gyro (CMG), Reaction jet Control System (RCS) controller effects to produce open and closed loop frequency domain as well as linear time domain response (via MATLAB). Any changes in control and filtering, can be fed back to the VAX through KERMIT to perform time-domain simulation.

Micro-Gravity Studies. Micro-gravity analysis is performed by measuring accelerations at one point on the Space station structure caused by forces and moments exerted at different points on the structure. SSSIM has a special provision for performing this analysis. You can select the points at which you want to compute the Micro-gravity accelerations, and you can locate actuators at the points where you want to insert the forces. Time history simulation can be run by connection function generators to the actuators.

5.0 APPLICATION TO SPACE STATION

Preliminary results are presented showing flex time history response and a linear controls design Nichols plot. An earlier Assembly Complete flex configuration is chosen as part of the validation process because of the availability of earlier test case data.

5.1 Simulated Space Station Flex Response

5.1.1 Time History Response

The objective is to simulate the reponse overall and to assess run time performance for a multibodied Space Station with a large number of flex models. The configuration is shown in Fig. 4 divided into 11 separate interconnected segments.

The core truss structure (body 1) is allowed 3 attitude rotations and 3 translations relative to LVLH. Each of the power booms (bodies 2, 3) undergoes a single rotation about a local 'y' axis (a gimbals) while each of the solar arrays (bodies 4 through 11) ar are kept locked (f gimbals). The NASTRAN flex models include: 12 'free-free' modes for the core, 12 'fixed-free' modes for each of the 2 power booms and 12 'fixed-free' modes for each of the 8 solar arrays. Damping ratios of 0.5% are assigned all modes. The simulation is run open-loop without aero drag or gravity. Excitation is by applying an initial 500 lb. pulse over 0.1s along the x axis.

Inertial 3-axis attitude rotations are shown in Fig. 5(a). They are, as expected, quite small (< 0.04 degrees) and there is an oscillatory perturbation induced by the combined flex effect. Generalized coordinates associated with the first three modes of body 1 are plotted in Fig. 5(b). The frequencies are quite low, especially when compared with mode 11 of body 11 (Fig. 5(c)). Overall the results

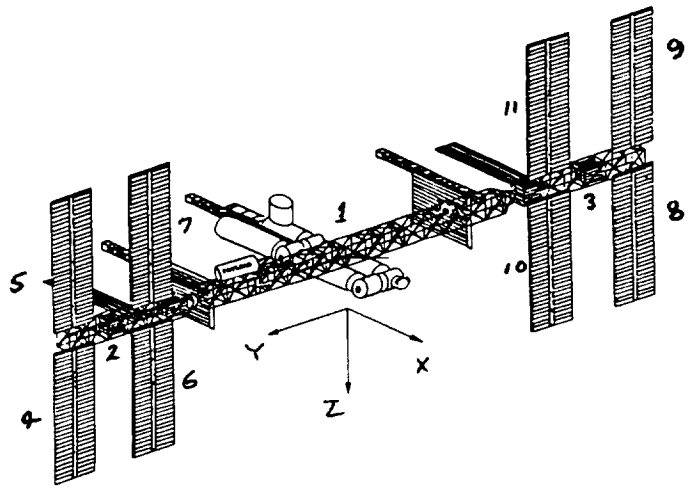


Fig. 4 Representative AssemblyComplete Configuration

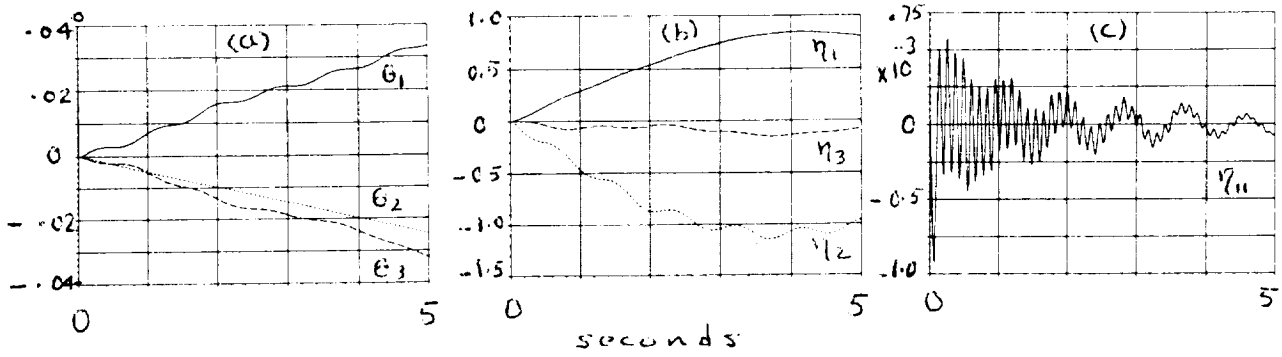


Fig. 5 Preliminary time history response demonstrating the flex capability of SSSIM Rev. 2.0: (a) attitude of core body relative to LVLH axes (b) generalized flex coordinate for first 3 modes of core body, and (c) generalized flex coordinate of mode 11 of solar array (body 11).

appear to be consistent and all indications are that the flex coupling effects have been captured. The same results are found using the TREETOPS software but much more computer CPU time is needed as discussed below.

5.1.2 Run Time Performance

The 5s simulation for the above case is carried out using a 4 term Runge-Kutta integrator at a step size of 0.025s. When run on a Sun Work Station SSSIM Rev. 2.0 requires 36 minutes as opposed to the 13 h ours needed to run TREETOPS. The improvement by a factor of at least 20 to 1 is dramatic enough to be evidence of a new era in multibody flex simulation. The improvement is almost as dramatic too by changing hardware to Digital Equipment VAX 6410 processor which completes this 140 DOF run in 3 minutes.

5.2 Linear Controls Analysis

SSSIM Rev. 2.0 also provides the linear models needed to carry out frequency response studies. For example, Fig. 6 shows a typical Nichols diagram generated for reaction jet control (RCS) along the roll (x) axis of an Assembly Complete configuration. The first order state estimator filter operates at 1 radian/s. All flex modes appear to be stable.

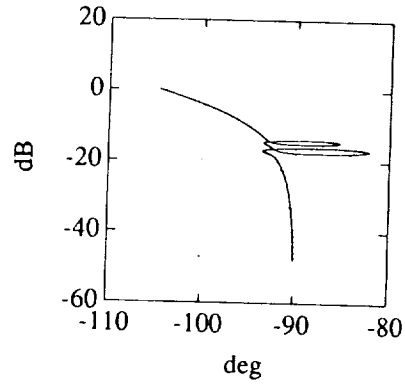


Fig. 4 Typical Assembly Complete 'roll' axis Nichols diagram for RCS.

6.0 CONCLUDING REMARKS AND FUTURE DEVELOPMENT

An order (n) nodal flex multibody model has been successfully developed for Space Station. The improvement in run time performance is significant, exceeding by at least an order of magnitude the existing generic codes (TREETOPS). To accommodate the flex modeling, an efficient preprocessor is built in which converts NASTRAN source data directly to the required input format and computes all additional rigid flex coupling coefficients.

A second major development is integration of the controlsdesign support programs into the SSSIM Rev. 2.0 environment. Analytic linearization and model reduction can now tie in directly with linear analysis software such as MATLAB.

Ongoing efforts include introduction of symbolic processing into solution of equations and into describing elements of the controller as well as of the environmental loads, such as gravity gradient. Alternate integration algorithms and strategies are under investigation, too, as possible means of improving run times.

REFERENCES

1. London, K., Singh, R., Schubele, B., "A Generic Multibody Dynamics, Control Simulation Tool for Space Station: SSSIM Rev. 1.1", 28th Aerospace Sciences Meeting, Paper No. AIAA 90-0745, Reno, NV, Jan. 8-111, 1990.

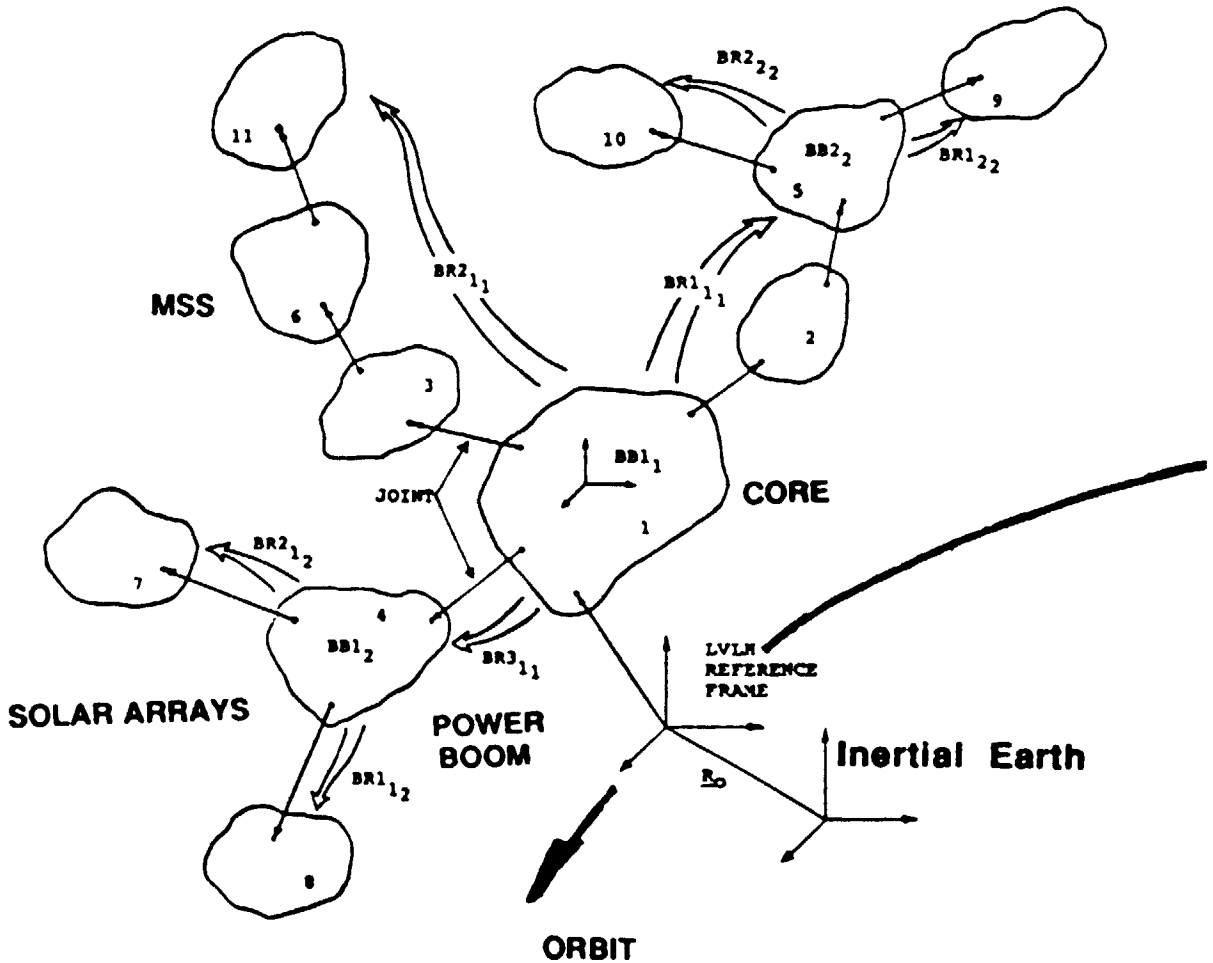
2. Singh, R.P., Schubele, D., and Sunkel, J.W., "Computationally Efficient Algorithm for the Dynamics of Multi-Link Mechanisms," AIAA Guidance, Navigation and Control Conference, Boston, MA, Aug. 14-16, 1989.

3. Space Station Simulation SSSIM Rev. 2.0 USERS MANUAL, DYNACS Engineering Co. Inc., Clearwater, FL, March 29, 1990.

4. Space Station Simulation SSSIM Rev. 2.0 Theory Manual, DYNACS Engineering Co. Inc., Clearwater, FL, March 29, 1990.

5. Singh, R.P., Vandervoort, R.J., and Likins, P.W., "Dynamics of Flexible Bodies in Tree Topology - A Computer-Oriented Approach." J. Guidance, Control and Dynamics, Vol 8. No. 5, Sept-Oct. 1985, pp. 584-590.

SSSIM REV. 2.0

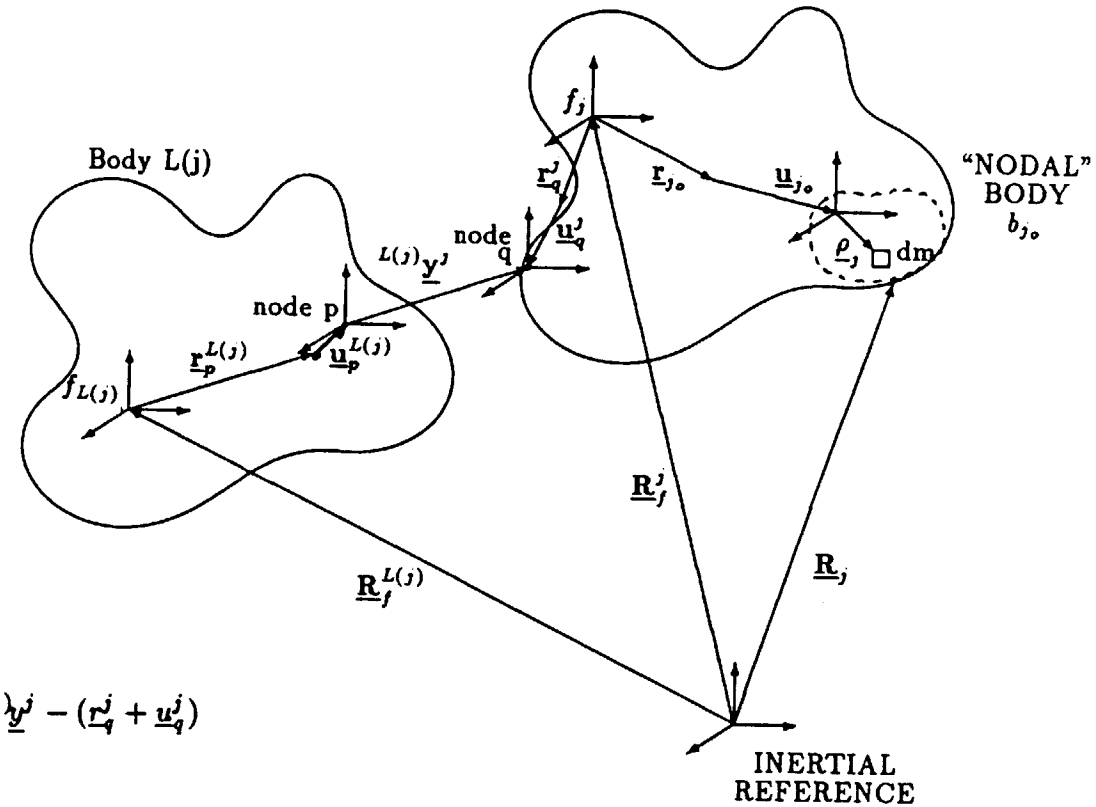


Generic multibody configuration with a multiple branch, multiple base body hierarchy, point-connected interbody joints and a prescribed orbiting LVLH reference applied to Space Station.

SSSIM REV. 2.0

$$\underline{u}_{jo} \triangleq \sum_{i=1}^{NMj} \underline{\phi}_i^j \, (\underline{r}_{jo}) \, \eta_i^j$$

$$\overset{o'}{\underline{u}}_{jo} \triangleq \sum_{i=1}^{NMj} \underline{\phi}_i^{'j} \, (\underline{r}_{jo}) \, \dot{\eta}_i^j$$

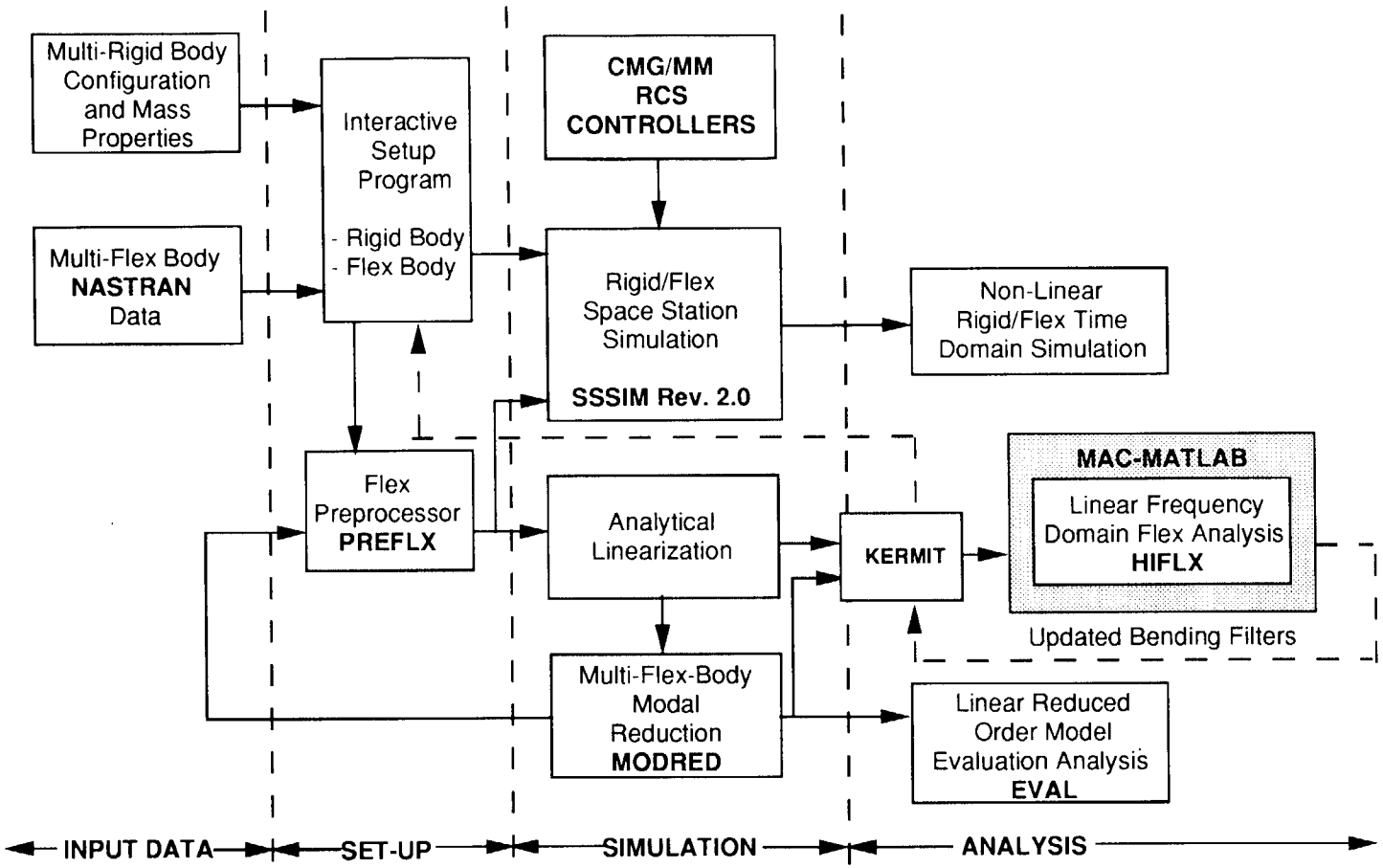


$$\underline{R}_j = \underline{R}_f^j + \underline{r}_{jo} + \underline{u}_{jo} + \underline{\rho}_j$$

$$\underline{R}_f^j = \underline{R}_f^{L(j)} + (\underline{r}_p^{L(j)} + \underline{u}_p^{L(j)}) + {}^{L(j)}\underline{y}^j - (\underline{r}_q^j + \underline{u}_q^j)$$

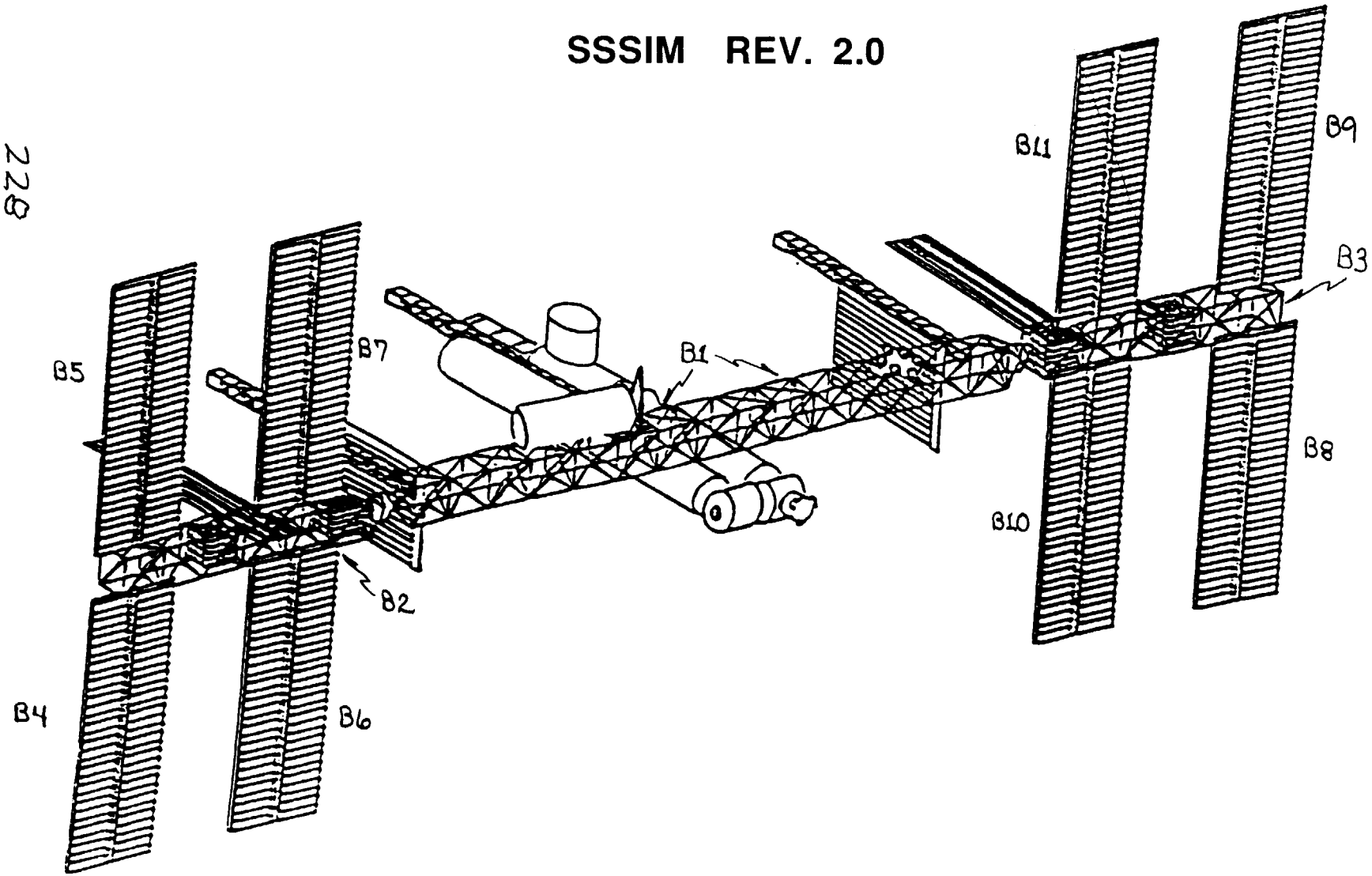
2 BODY GEOMETRY FORMS BASIS FOR RECURSIVE KINEMATICS:
[L(j) = INBOARD
j = OUTBOARD]

SSSIM REV. 2.0



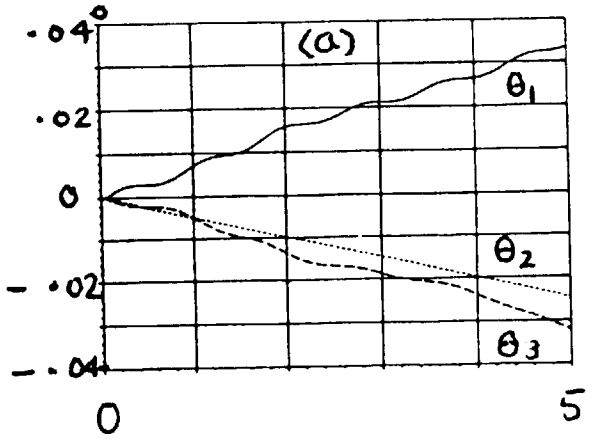
INTEGRATED MULTI-RIGID/FLEX-BODY DYNAMICS, CONTROLS SIMULATION, ANALYSIS CAPABILITY

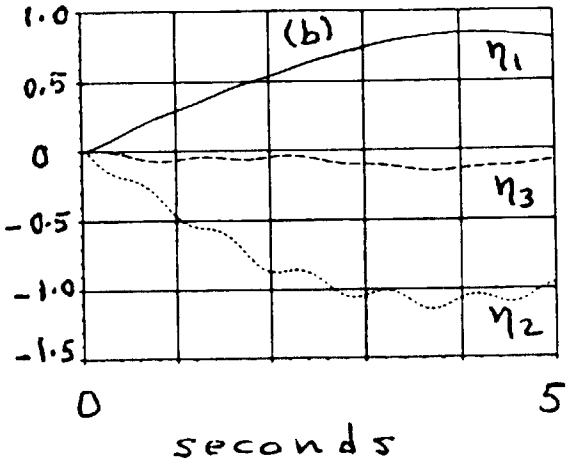
SSSIM REV. 2.0

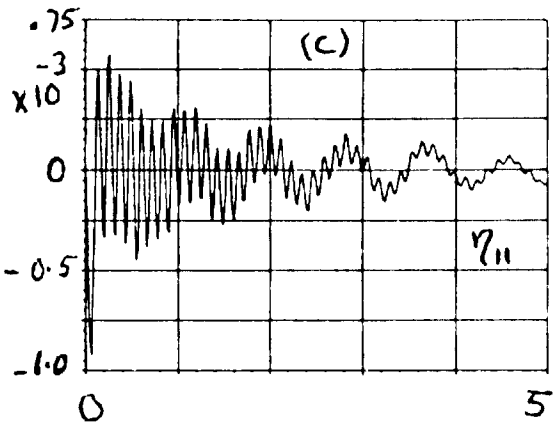


II-BODY BREAKDOWN FOR ASSEMBLY COMPLETE SPACE STATION TEST CASE

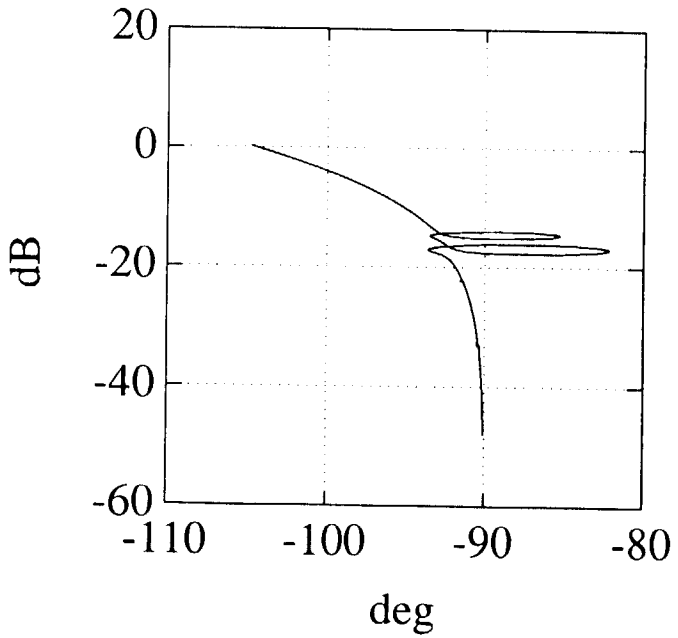
SSSIM REV. 2.0







SSSIM REV. 2.0 RESPONSE FOR II-FLEX BODY SPACE STATION (a) CORE BODY ATTITUDE; (b) CORE BODY GENERALIZED FLEX COORDINATES; AND (c) BODY II (SOLAR ARRAY) MODE II GENERALIZED FLEX COORDINATE.



NICHOLS DIAGRAM FOR RCS ROLL ('X') AXIS CONTROL OF ASSEMBLY COMPLETE CONFIGURATION

An Integrated Control/Structure Design Method using Multi-objective Optimization

Sandeep Gupta

ViGYAN, Inc.
30 Research Drive,
Hampton, VA 23666

and

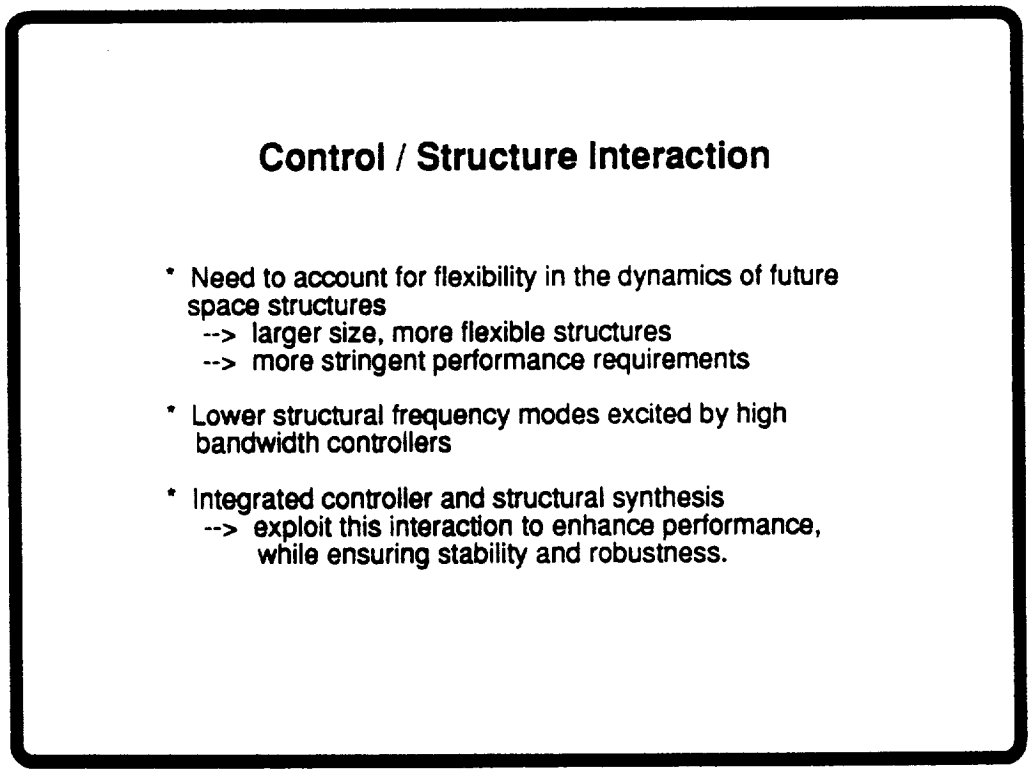
Suresh M. Joshi

NASA Langley Research Center
Hampton, VA 23665

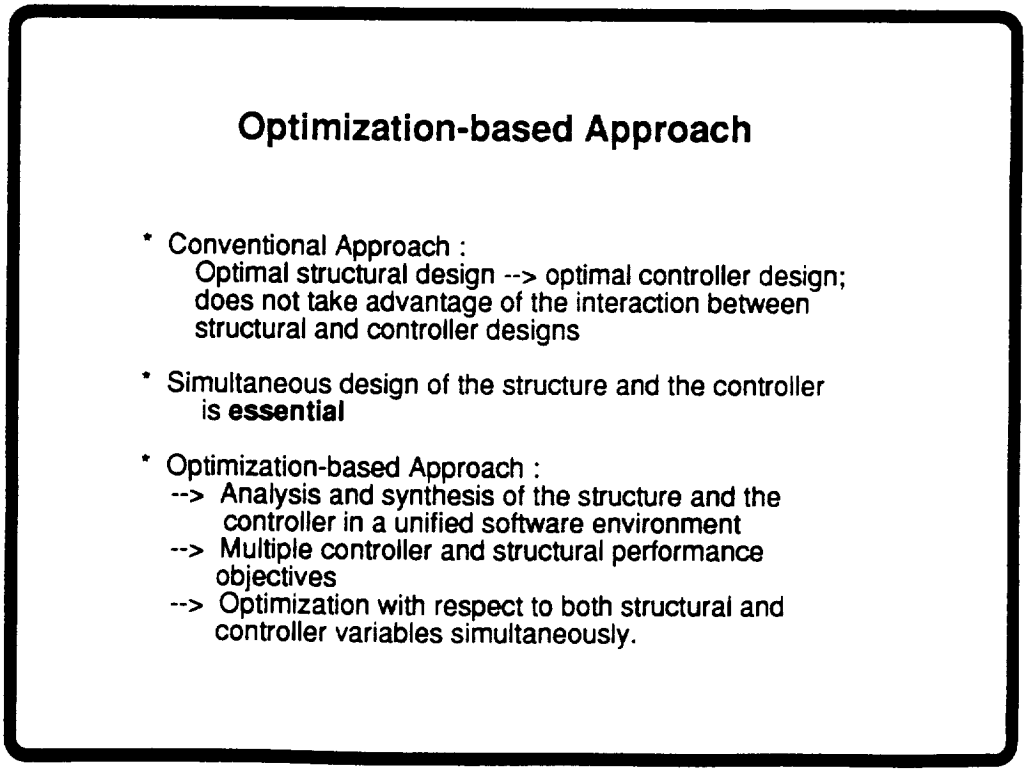
Presented at the
4th NASA Workshop on Computational Control
of Flexible Aerospace Systmes.

Kingsmill Resort, Williamsburg, Virginia.
July 11-13, 1990.

Control/structure interaction is an issue in some current space structures like the RMS arm on the shuttle, and will definitely be an issue in the design of future spacecraft which will be larger in size, and therefore more flexible. At the same time, these structures would have to meet more stringent performance requirements so that a rigid-body approximation of the structural dynamics would be inadequate. Low structural frequencies of these large, flexible structures will fall within the bandwidth of the controllers used to control these structures. This would lead to significant interaction between the control inputs and the flexible structural response. Hence, integrated controller and structural synthesis is necessary to enhance system performance while ensuring stability and robustness of the system.



The conventional approach has been to design an optimal structure, and then design an optimal controller for that structure. However, this approach does not take advantage of the interaction between the structural and the controller design processes. Note that in contrast to the physical interaction between control inputs and flexible structural response, here we are referring to the interaction during the synthesis process. Hence, simultaneous design of the controller and the structure is essential for the optimal synthesis of the entire system. The optimization-based approach is to perform analysis and synthesis of the structure and the controller in a unified software environment (as opposed to using separate tools for structural analysis and control design) and to optimize the structural and controller objectives with respect to both structural and controller variables. With numerous structural and controller objectives to be optimized, this approach naturally leads to multi-objective optimization.



Optimization of a single objective function, $f(\mathbf{x})$, sometimes referred to as scalar performance index, over a constrained design space, Ω, is mathematically well-defined, since real numbers have a unique and complete order. However, this is not the case for multi-objective optimization. The objectives, $f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_m(\mathbf{x})$, can be considered as the elements of a vector, $\mathbf{f}(\mathbf{x})$, in the criteria space $\mathbf{R}^m$; therefore this problem is sometimes called optimization of a vector performance index. However, minimization of a vector is meaningless, since the criteria space, $\mathbf{R}^m$, does not have any unique and complete order. A partial order of the space $\mathbf{R}^m$, called the "natural order", leads to the concept of Pareto optimality [1].

Multi-objective Optimization

Scalar performance index :

$$\min_{\mathbf{x} \in \Omega} \quad f(\mathbf{x})$$

where $\Omega = \{ \mathbf{x} \in \mathbf{R}^n \mid u_i(\mathbf{x}) \leq 0, i = 1,2,..,p;\ v_j(\mathbf{x}) = 0, j = 1,2,...,q \}$

---> well defined, since real numbers have unique, complete order

Vector performance index :

$$\text{"min"}_{\mathbf{x} \in \Omega} \quad \mathbf{f}(\mathbf{x}) \qquad \text{where } \mathbf{f}(\mathbf{x}) = \{ f_1(\mathbf{x}),\ f_2(\mathbf{x}),\ ...,\ f_m(\mathbf{x}) \}^T$$

---> no unique, complete order for $\mathbf{R}^m$

---> a partial order of space $\mathbf{R}^m$, called " natural order ", leads to the concept of Pareto optimality

The definition of Pareto optimality is given below. A vector, $\mathbf{x}^*$, is Pareto optimal for a multi-objective optimization if and only if there exists no feasible vector $\mathbf{x}$ such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for all i, and $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$ (with a strict inequality) for at least one j. In words, $\mathbf{x}^*$ is Pareto optimal if there exists no feasible vector $\mathbf{x}$ that would decrease some objective function $f_j(\mathbf{x})$ without causing a simultaneous increase in some other objective function. Conversely, if $\mathbf{x}$ is not a Pareto optimal vector, some objective $f_j(\mathbf{x})$ can be decreased without increasing any other objective (other objectives decrease or remain constant).

Pareto Optimality

A vector $\mathbf{x}^*$ is Pareto optimal for the multi-objective optimization problem

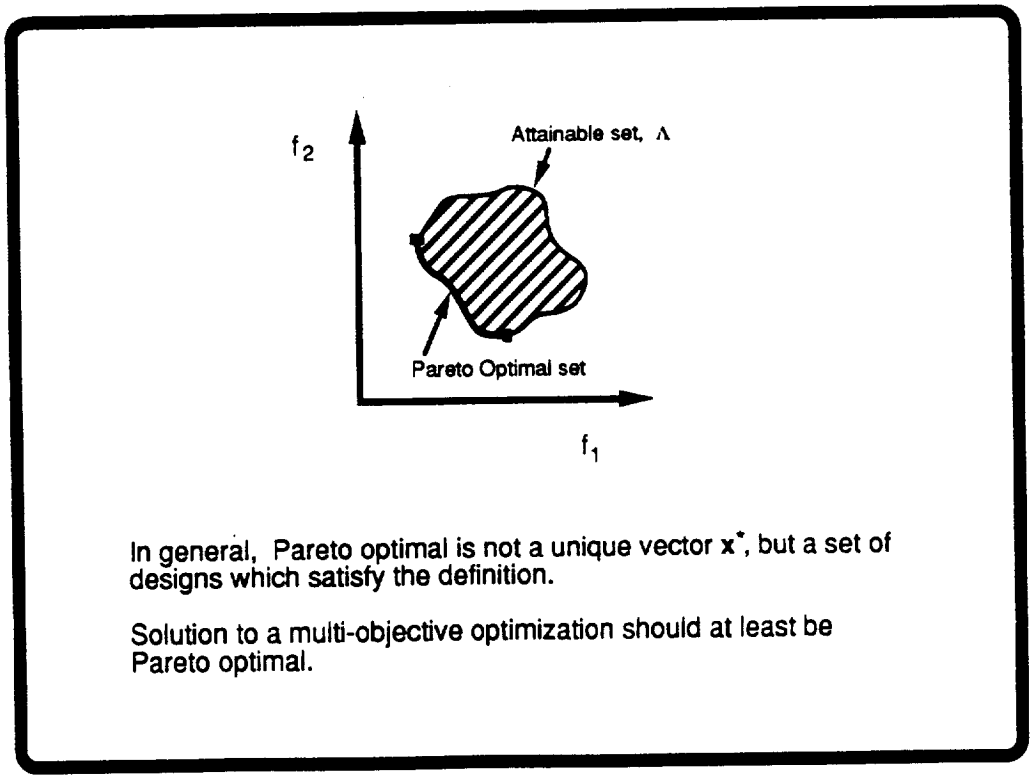
$$\text{"min"}_{\mathbf{x} \in \Omega} \quad \mathbf{f}(\mathbf{x}) \qquad \text{where } \mathbf{f}(\mathbf{x}) = \{ f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}) \}^T$$

if and only if there exists no $\mathbf{x} \in \Omega$ such that

$$f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*) \quad \text{for } i = 1, 2, \ldots, m, \text{ and}$$
$$f_j(\mathbf{x}) < f_j(\mathbf{x}^*) \quad \text{for at least one } j.$$

In words, $\mathbf{x}^*$ is Pareto optimal if there exists no feasible vector $\mathbf{x}$ that would decrease some objective $f_i(\mathbf{x})$ without causing a simultaneous increase in at least one other objective.

This figure graphically shows Pareto optimal solutions for a two objective optimization problem. The shaded area is the attainable set, Λ, which is the image of the feasible design space, Ω, under the transformation of the objective functions, $\mathbf{f}(\mathbf{x})$, in the criteria space $\mathbf{R}^2$. The thick line shows the Pareto optimal set. Along this line we cannot decrease one objective function without causing a simultaneous increase in the other [1]. As seen here, in general, Pareto optimal is not a unique vector, but a set of vectors which satisfy the definition. The designer still has ample choice to select a good design after performing parametric trade-off studies among different objectives, but a solution to the multi-objective optimization should at least be Pareto optimal.



The approach used here for multi-objective optimization follows from the goal attainment approach [2]. For this approach, a set of functions, $g_j(\mathbf{x})$, are derived from the objective functions, $f_j(\mathbf{x})$, as shown below. The parameters a_j are certain reference values or goal values of the objectives, and b_j are scaling parameters so that various objectives become commensurable, and some weights are assigned to the objectives. As seen from the expression for $g_j(\mathbf{x})$, the weight of each objective depends inversely on b_j. A solution to the multi-objective optimization problem is then given by minimizing over $\mathbf{x}$ the maximum of g_j's. Problems with this approach are that the maximization function is non-differentiable, which disrupts most numerical nonlinear programming algorithms. Also, the solution from this approach may not be Pareto optimal.

Goal Attainment Approach for Multi-objective Optimization

$$g_j(\mathbf{x}) = \frac{f_j(\mathbf{x}) - a_j}{b_j} \qquad \text{for } j = 1,2,\ldots,m,$$

where a_j are reference values or goal values,
b_j are scaling parameters so that various objectives are commensurable.

$$\min_{\mathbf{x}} \left[\max_{j} \{ g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_j(\mathbf{x}), \ldots, g_m(\mathbf{x}) \} \right]$$

Problems:
1) nondifferentiable max function
2) solution may not be Pareto optimal

The Kreisselmeier-Steinhauser (KS) function [3,4] given by the expression below provides an approximation for the maximum of a set of functions $g_j(\mathbf{x})$. Values of the KS function remain within the interval $g_{max} \leq \text{KS} \leq g_{max} + (\ln m)/\rho$, where g_{max} is the maximum of the set and ρ is a positive scalar parameter of the KS function. For larger values of ρ, the KS function provides a closer approximation of the maximization function. Secondly, the KS function is differentiable with respect to the design variable, x_i, as given by the expression below. This assumes that the functions $g_j(\mathbf{x})$ are differentiable with respect to x_i, which follows from the differentiability of the objective functions $f_j(\mathbf{x})$. Thus, the KS function provides a differentiable approximation for the maximization function. The first expression for the KS function tends to cause floating point overflow problems in numerical computations. An algebraically equivalent expression for the KS function is given below to avoid this problem.

Kreisselmeier-Steinhauser (KS) Function

$$KS = \frac{1}{\rho} \ln \left\{ \sum_j \exp \left[\rho g_j(\mathbf{x}) \right] \right\}$$

Useful properties:

1) $g_{max}(\mathbf{x}) \leq KS \leq g_{max}(\mathbf{x}) + (\ln m)/\rho$

2)
$$\frac{\partial KS}{\partial x_i} = \frac{\sum_j \exp\{\rho g_j(\mathbf{x})\} \frac{\partial g_j(\mathbf{x})}{\partial x_i}}{\sum_j \exp\{\rho g_j(\mathbf{x})\}}$$

KS with large ρ provides a differentiable approximation for the function $\max_j \{ g_j(\mathbf{x}) \}$

$$KS = g_{max} + \frac{1}{\rho} \ln \left\{ \sum_j \exp \left[\rho \left(g_j(\mathbf{x}) - g_{max} \right) \right] \right\}$$

has good computational properties.

Our approach to multi-objective optimization is to use the KS function instead of the maximization function in the goal attainment approach. The expression given here for $g_j(\mathbf{x})$ corresponds to an objective function which has to be minimized. In case an objective has to be maximized, we use the standard approach of minimizing the negative of $g_j(\mathbf{x})$ instead. Also, there is no penalty for over-achieving the goals, i.e., if all the reference values have been reached, the optimization algorithm is free to take the g_j's to negative values. Minimization over $\mathbf{x}$ of the KS function of $g_j(\mathbf{x})$ gives a multi-objective optimal solution. Thus, with this approach the problem is reduced to an unconstrained minimization (apart from side constraints on the design variables) of a differentiable function, which is very amenable to numerical nonlinear programming algorithms. Furthermore, the minimum KS solution is a Pareto optimal solution.

Multi-objective Optimization Approach

$$\min_{\mathbf{x}} \; KS$$

$$\min_{\mathbf{x}} \; \left\{ g_{max} + \frac{1}{\rho} \ln \left(\sum_j \exp \left[\rho \left(g_j(\mathbf{x}) - g_{max} \right) \right] \right) \right\}$$

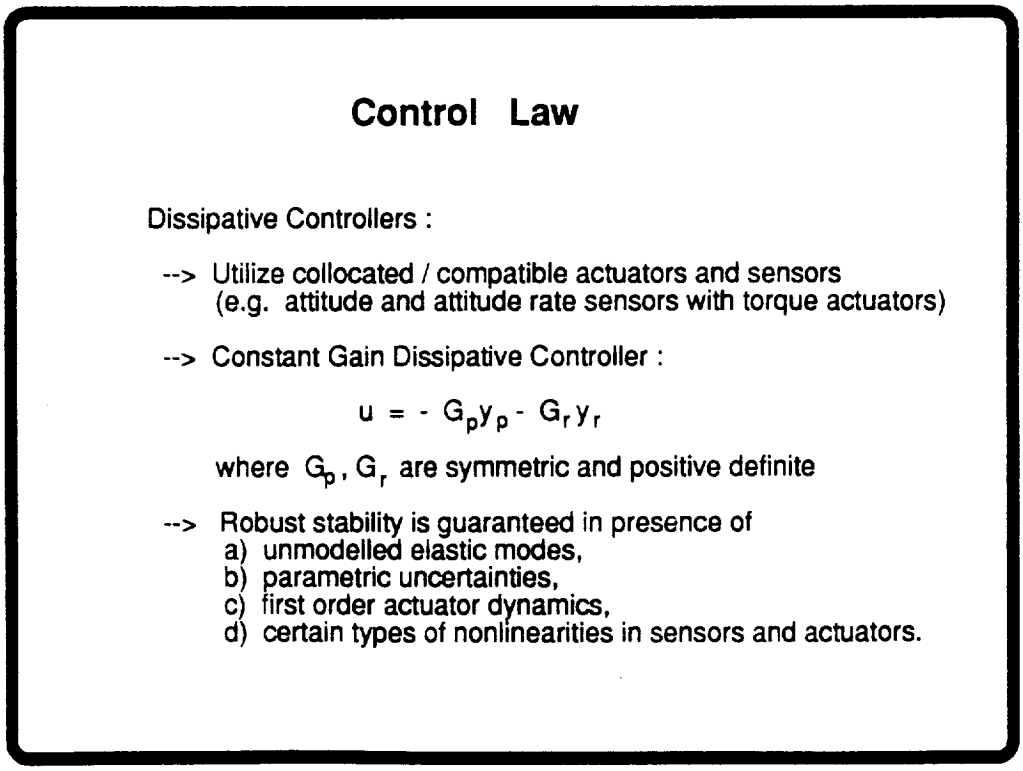
where $$g_j(\mathbf{x}) = \frac{f_j(\mathbf{x}) - a_j}{b_j} \qquad \text{for } j = 1,\ldots,m$$

gives a multi-objective optimal solution.

--> unconstrained minimization of a differentiable function

--> min KS solution is a Pareto optimal solution.

Structural analysis for CSI design is performed using finite element modeling of the structure, i.e., assembling the mass and stiffness matrices of the discretized structure and solving a generalized eigenvalue problem to determine a desired number of natural frequencies and mode shapes of the structure. With some assumed modal damping, the structural dynamics equations in modal form are used as open-loop plant dynamics for control design. The feedback control law used herein is a dissipative controller employing collocated and compatible pairs of actuators and sensors [5,6]. For example, torque actuators are used with attitude and attitude rate sensors. A constant gain dissipative feedback controller is presented here. The gain matrices, G_p and G_r, have to be positive definite to ensure the dissipative properties of the controller. With these properties, the controller provides guaranteed robust stability in the presence of unmodelled elastic modes, parametric uncertainties (in modal frequencies, mode shapes etc.), first-order actuator dynamics, and certain types of nonlinearities in the actuators and sensors [6].



The open-loop plant model has a large number of structural modes, and a reduced-order model is needed for control/strucutred integrated design. We use a novel H_∞ norm-based approach for this model reduction. The idea is to include the modes which are most significant in the overall system transfer function. The system transfer function is the rigid body transfer function, $G_{RB}(s)$, plus a sum of the components of the transfer function due to each flexible mode, $G_{ei}(s)$, as shown below. The contribution of each mode to the overall transfer function is measured by the H_∞ norm of $G_{ei}(s)$ given by the expression below. For the reduced order model, we pick the modes with the largest H_∞ norms of $G_{ei}(s)$.

Model Reduction

--> H_∞ norm-based approach to select modes for reduced order model

System transfer function is

$$Y(s) = \left[\frac{T J^{-1} T^T}{s^2} + \sum_{i=1}^{n} \frac{(\phi_i \phi_i^T)}{s^2 + 2\rho_i \omega_i s + \omega_i^2} \right] U(s)$$

(Rigid body) (Flexible structure)

$$G(s) = G_{RB}(s) + \sum_{i=1}^{n} G_{ei}(s)$$

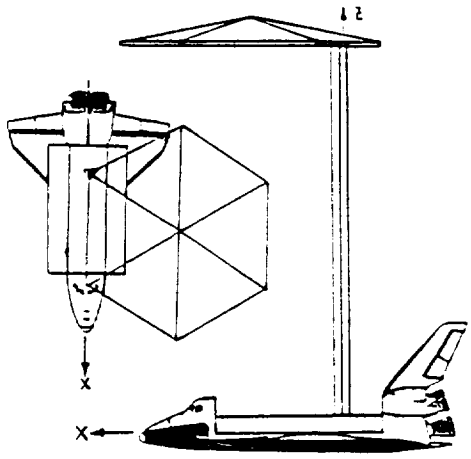
contribution of each mode to transfer function is measured by H_∞ norm of $G_{ei}(s)$.

$$\| G_{ei}(s) \|_\infty = \frac{(\phi_i^T \phi_i)}{2 \omega_i^2 \rho_i (1-\rho_i^2)^{\frac{1}{2}}}$$

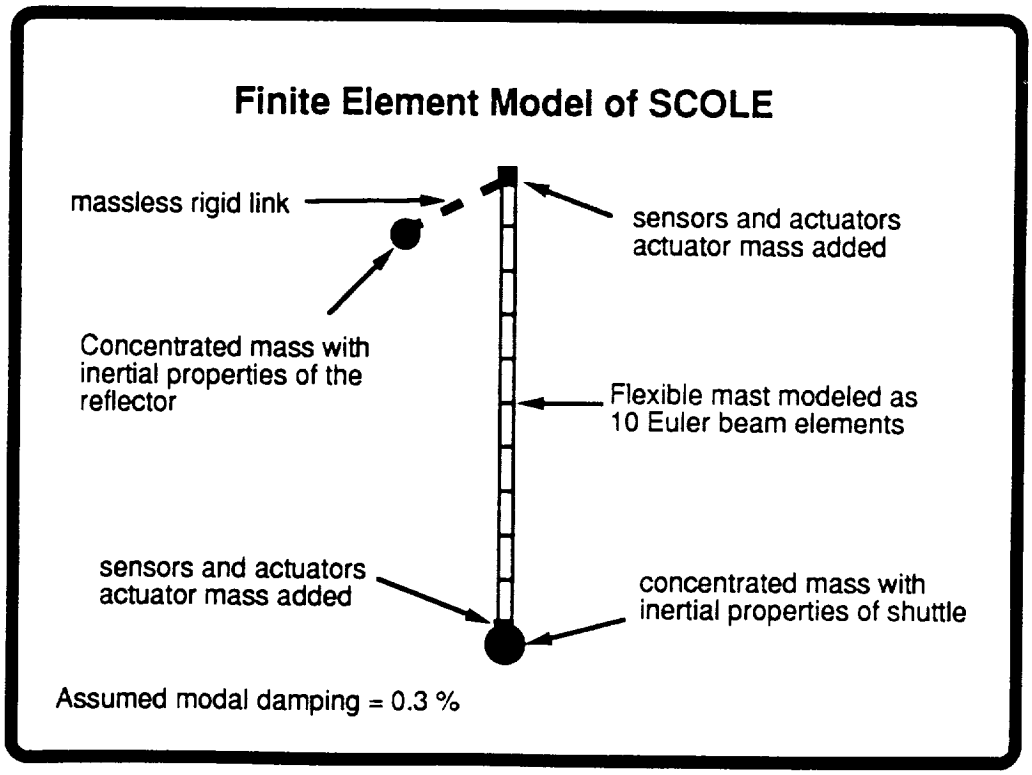
--> pick modes with largest $\| G_{ei}(s) \|_\infty$

In order to demonstrate this control/structure integrated design approach, an integrated design of the Spacecraft Control Laboratory Experiment (SCOLE) configuration [7] was performed. The SCOLE consists of a reflector antenna attached to the shuttle by a flexible mast. This configuration was chosen because of its structurally simple model as a flexible beam with off-set, inertial massess at either end.

Example : CSI Redesign of SCOLE



A finite element model of SCOLE used in this study is shown here. The flexible mast is modelled as 10 Euler beam elements. The space shuttle is represented by a concentrated mass at the tip with inertial properties (including mass, moments of inertia and products of inertia) of the shuttle. The off-set reflector is modelled as a concentrated mass with inertial properties of the reflector at its center of mass, which is connected to the tip of the mast by a massless rigid link. Compatible and collocated pairs of actuators and sensors are placed at the tips of the flexible mast. The actuator masses depend on the control gains and are included in the finite element model. A modal damping of 0.3% is assumed. Numerical values of various parameters were taken from a paper by Poelaert [8].



Structural design variables are the diameters of tubular beam elements. Hence, there are 10 structural design variables. Controller design variables determine the attitude and attitude rate gain matrices for the constant gain dissipative feedback control law. Since G_p and G_r have to be positive definite for this controller, we use Cholesky factors of the gain matrices as follows: $G_p = L_p L_p^T$, and $G_r = L_r L_r^T$, where L_p and L_r are lower triangular matrices. The controller design variables are the elements of L_p and L_r matrices. Subsequently, there are 24 controller design variables in this study.

Design Variables

Structural:

10 structural design variables -- diameters of tubular beam elements

Controller:

Controller design variables -- the attitude and rate gain matrices
Since G_p, G_r, are positive definite, we use Cholesky factors for design

$G_p = L_p L_p^T$ and $G_r = L_r L_r^T$

Hence, controller design variables are elements of L_p and L_r

24 controller variables for this case.

Four control/structure objectives were selected for this study. The first structural objective is to minimize mass of the flexible mast and the actuators, M_T; obtained by subtracting the fixed masses of the shuttle and the reflector from the total mass of the system. Another structural objective is to maximize the first open-loop frequency, ω_1, so that the structure can be made as stiff as possible within allowable values of the mass and the control objectives. The first control objective is to minimize a measure of transient response decay time, τ. This measure is the sum of the reciprocal of absolute real parts of the closed-loop eigenvalues, as shown below. The last CSI objective is to minimize a noise attenuation measure, σ, which is the steady-state root mean square attitude error due to a white noise input at the sensors.

CSI Performance Objectives

Structural:

1) Mass of flexible mast and actuator mass, M_T
2) First open-loop structural frequency, ω_1

Controller:

3) A measure of transient response decay time, τ

$$\tau = \sum_{i=1}^{n} \frac{1}{|\ \mathrm{Re}\ \lambda_i\ |}$$

4) A noise attenuation measure -- root mean square attitude error due to a white noise input at the sensors, σ

The main issue in using approaches which employ incremental scaling parameters a_j and b_j is to select values for these parameters. Although arbitrary values of these parameters would still lead to a Pareto optimal solution with our approach, some preliminary optimizations are performed to establish trends in the behavior of various objectives, which assist in choosing reasonable values for the scaling parameters. Minimizing mass, M_T, alone takes the structural variables to their lower bounds (for least possible structural mass), and the controller variables close to zero (for least actuator mass). However, this makes the structure very flexible. Maximizing first open-loop frequency, ω_1, sends the structural variables to their upper bounds and the controller variables close to zero (near zero point masses at the tips). This leads to a very massive and stiff structure. These optimizations show the tradeoff between the first two objectives. Optimizing both, i.e. minimizing mass and maximizing the first open-loop frequency, simultaneously, results in reasonable values for both objectives.

Trends from Optimization

* Minimizing mass M_T :
  structural variables --> lower bounds
  controller variables --> close to zero
  Mass $M_T \sim 10^{-4}$
  very flexible structure, $\omega_1 \sim 10^{-3}$; $\tau \sim 10^{9}$; $\sigma = 1.65 \times 10^{-5}$

* Maximizing first open loop frequency ω_1 :
  structural variables --> upper bounds
  controller variables --> close to zero
  First open loop frequency $\omega_1 = 12.24$
  very massive structure, $M_T \sim 15300$, $\tau = 134.26$, $\sigma = 2.3 \times 10^{-5}$

* Structural optimization -- minimizing mass, maximizing frequency :
  structural variables -- thick close to shuttle end, thin out at the reflector end
  controller variables -- close to zero
  $M_T = 2827.3$, $\omega_1 = 0.173$,
  $\tau = 99.9$, $\sigma = 2.15 \times 10^{-5}$

The previous optimizations were primarily structural optimizations. In controls, first a rigid body controller is designed (with no optimization) as follows: $G_p = \omega^2 J$ and $G_r = 2\rho\omega J$, where J is the moments of inertia matrix for the structure and ρ, ω are closed-loop damping ratio and frequency. With $\rho = 0.707$ and $\omega = 0.05$ for the nominal SCOLE configuration, we get $\tau = 121.34$ and $\sigma = 2.16 \times 10^{-5}$. Next, an optimization is performed with respect to control variables only, while using the nominal structural configuration for SCOLE, resulting in reduction of τ to 104.37. Finally, performing simultaneous optimization with respect to structural and controller variables reduces τ further to 65.86, but this results in a massive and stiff structure with $M_T = 5537.6$, since there was no restriction on mass. Thus, there is another tradeoff involved between mass, M_T, and the transient response decay measure, τ.

Minimizing transient response decay time measure, τ :

1) No optimization. Using rigid body controller as
   $G_p = \omega^2 J$ and $G_r = 2\rho\omega J$
   where $\omega = 0.05$ and $\rho = 0.707$
   $\tau = 121.34$ and $\sigma = 2.16 \times 10^{-5}$

2) Optimization with controller variables only.
   structural variables at nominal value
   $\tau = 104.37$ and $\sigma = 2.12 \times 10^{-5}$

3) Optimization with both structural and controller variables
   $\tau = 65.86$ and $\sigma = 2.16 \times 10^{-5}$
   but since there is no restriction on mass, $M_T = 5537.67$

With some idea of the tradeoffs among different objectives and some insight into the numerical values of the objectives involved for selecting the parameters a_j and b_j, multi-objective optimizations are performed, next. The control-optimized design is used as the initial design. In order to reduce both mass and response decay time, lower values of desired mass and desired response decay time are used. The desired values for the first open-loop frequency and the noise attenuation measure, are used more as constraint values than performance objectives. The parameters b_j were chosen to make the incremental variations from the desired values commensurable. The optimization results in lower values for both mass, $M_T = 2847.7$, and response decay time measure, $\tau = 94.622$. However, the first open-loop frequency is lower than its desired value. To emphasize this objective more, we reduce the value of parameter b_2. Now, the optimal first open-loop frequency is much closer to its desired value; but the mass and the response decay time measure, are not reduced quite as much.

Multi-objective Optimization Results

	M_T	ω_1	τ	σ
INITIAL	2881.0	0.2055	104.37	2.1×10^{-5}
a_j^1	2800.0	0.2	90.0	2.2×10^{-5}
b_j^1	10.0	0.01	1.0	10^{-4}
f_{opt}^1	2847.7	0.153	94.622	2.14×10^{-5}
a_j^2	2800.0	0.2	90.0	2.2×10^{-5}
b_j^2	10.0	0.001	1.0	10^{-4}
f_{opt}^2	2880.6	0.197	97.92	2.21×10^{-5}

Previous results demonstrated that mass was the constraining factor in improving the response decay time. In fact, the optimizer was reducing the structural mass making the mast more flexible and adding the mass to the actuators to improve τ. Therefore, in the next series of optimizations, the desired mass is increased to allow reduction in response decay time. Optimization resutls indeed show the trend of the mass going up while τ is reduced. In a similar manner, by varying the values of a_j and b_j, the designer can place different emphasis on various objectives, and perform parametric tradeoff studies with Pareto optimal designs.

Multi-objective Optimization Results

	M_T	ω_1	τ	σ
INITIAL	2881.0	0.2055	104.37	2.1×10^{-5}
a_j^3	3000.0	0.2	90.0	2.2×10^{-5}
b_j^3	10.0	0.01	1.0	10^{-4}
f_{opt}^3	2996.11	0.292	89.74	2.15×10^{-5}
a_j^4	3500.0	0.2	80.0	2.2×10^{-5}
b_j^4	10.0	0.01	1.0	10^{-4}
f_{opt}^4	3575.2	0.485	87.53	2.15×10^{-5}

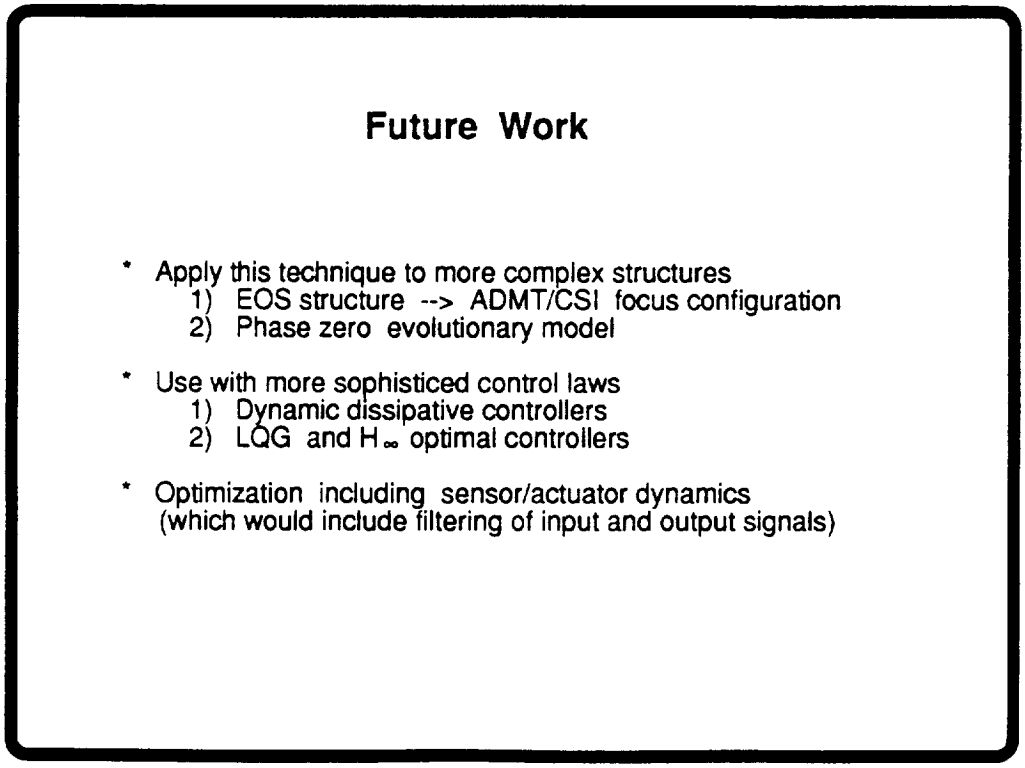
This paper demonstrates the benefits of a multi-objective optimization-based control/structure integrated design methodology. An application of the proposed CSI methodology to the integrated design of the SCOLE configuration is presented here. Integrated design resulted in reducing both the control performance measure, τ, and the mass, M_T. Thus, better overall performance is achieved through integrated design optimization.

The multi-objective optimization approach used here provides Pareto optimal solutions by unconstrained minimization of a differentiable KS function. Furthermore, adjusting the parameters a_j and b_j gives insight into the trade-offs involved between different objectives.

Concluding Remarks

* Control/Structure Integrated Design:
  - --> Example application of CSI methodology to SCOLE
  - --> Integrated design optimization gives better overall performance
* Multi-objective Optimization Approach :
  - --> Pareto optimal solutions
  - --> Unconstrained optimizations (constraints can be included as desired values with large weights)
  - --> Adjusting a_j and b_j gives insight into tradeoffs involved between different objectives

Benefits of the CSI design were observed in a structurally simple SCOLE configuration. Greater opportunity for such benefits exists in the CSI design of more complex space structures. We will be applying this methodology to: 1) the EOS (Earth Observing System) structure, which is the ADMT/CSI focus configuration, and 2) the phase-zero evolutionary model at NASA Langley Research Center. This methodology will also be used with more sophisticated control laws such as dynamic dissipative controllers, as well as, LQG and H_∞ optimal controllers. Also, open-loop plant dynamics could be refined by including sensor/actuator dynamics (which would include filtering of input and output signals).



References

1. Schy, A. A. and Giesy, D. P.: *Multicriteria Optimization Methods for Design of Aircraft Control Systems,* pp 225-262, Multicriteria Optimization in Engineering and in the Sciences, Plenum Press, New York, 1988.

2. Gembicki, F. W. and Haimes, Y. Y.: *Approach to Performance and Sensitivity Multiobjective Optimization: The Goal Attainment Method,* pp 769-771, IEEE Transactions on Automatic Control, December 1975.

3. Kreisselmeier, G. and Steinhauser, R.: *Systematic Control Design by Optimizing a Vector Performance Index,* pp 113-117, Computer Aided Design of Control Systems, Zurich, Switzerland, August 1979.

4. Sobieszczanski-Sobieski, J., Dovi, A. R. and Wrenn, G. A.: *A New Algorithm for General Multiobjective Optimization,* NASA Technical Memorandum 100536, March 1988.

5. Joshi, S. M. and Maghami, P. G.: *Dissipative Compensators for Flexible Spacecraft Control,* Proceeding of 1990 American Control Conference, San Diego, CA, May 23-25, 1990.

6. Joshi, S. M.: *Control of Large Flexible Space Structures,* Vol. 131, Lecture Notes in Control and Information Sciences, Springer-Verlag, Berlin, 1989.

7. Taylor, L. W. and Balakrishnan, A. V.: *A Mathematical Problem and a Spacecraft Control Laboratory Experiment (SCOLE) Used to Evaluate Control Laws for Flexible Spacecraft ... NASA/IEEE Design Challange,* pp 2-28, SCOLE Workshop, Hampton, December 1984.

8. Poelaert, D.: *SCOLE Dynamic Analysis and Frequency Response using DISTEL Software,* pp 41-86, SCOLE Workshop, Hamtpon, December 1984.

COMBINED STRUCTURES-CONTROLS OPTIMIZATION OF LATTICE TRUSSES

A. V. Balakrishnan

Research supported in part under NASA Task Assignment 49: NAS1-18585.

COMBINED STRUCTURES-CONTROLS OPTIMIZATION OF LATTICE TRUSSES

A. V. Balakrishnan

ABSTRACT

The purpose of this paper is to demonstrate concretely the role that distributed parameter models can play in CSI, in particular in combined structures-controls optimization problems of importance in preliminary design. Closed form solutions can be obtained for performance criteria such as rms attitude error, making possible analytical solutions of the optimization problem. This is in contrast to the need for numerical computer solution involving the inversion of large matrices in traditional finite element model use. Another advantage of the analytic solution is that it can provide much needed "insight" into phenomena that can otherwise be obscured — or difficult to discern from numerical computer results.

As a compromise in level of complexity between a toy laboratory model and a real space structure we have chosen the lattice truss used in the EPS (Earth Pointing Satellite). The optimization problem chosen is a generic one: of minimizing the structure mass subject to a specified stability margin and to a specified upper bound on the rms attitude error ("tip response"), using co-located controller and sensors. Standard FEM treating each bar as a truss element is used, while the continuum model is anisotropic Timoshenko beam model. Performance criteria are derived for either model, except that for the distributed parameter model we obtain explicit closed form solutions. Numerical results obtained by the two models show complete agreement. Based on the continuum model we obtain a solution to the problem of optimal placement of actuators to minimize mean square attitude error. A canonical optimization problem is examined and shown to be trivial, and even capable of analytical solution, using the continuum model performance criteria formulas in contrast to the complex computer solutions based on FEM or truncated modal models currently in vogue.

Introduction

The most voiced criticism against the use of continuum models for structures is that they are (a) impossible to derive for a realistic structure and (b) even if it could be done, calculations using the model are equally impossible. We shall show that both statements are false — at least in the CSI optimization problem — in particular in preliminary design.

In combined controls-structures optimization, the optimization is the least difficult — the real challenge is to derive expressions for the chosen performance criteria in terms of the controls/structures parameters. We shall show that such formulas are much simpler when continuum models are used — moreover in many cases we can derive explicit closed form expressions in terms of elementary function which can actually trivialize the optimization problem. In particular the techinques of optimization need no longer dominate.

The purpose of this paper is to demonstrate concretely the role that distributed parameter models can play in CSI, in particular in combined structures-controls optimization problems of importance in preliminary design. Closed form solutions can be obtained for performance criteria such as rms attitude error, making possible analytical solutions of the optimization problem. This is in contrast to the need for numerical computer solution involving the inversion of large matrices in traditional finite element model use. Another advantage of the analytic solution is that it can provide much needed "insight" into phenomena that can otherwise be obscured — or difficult to discern from numerical computer results.

As a compromise in level of complexity between a toy laboratory model and a real space structure we have chosen the lattice truss used in the EPS (Earth Pointing Satellite). This is described in Section 1. The optimization problem chosen is a generic one: of minimizing the structure mass subject to a specified stability margin and to a specified upper bound on the rms attitude error ("tip response"). The mathematical statement of

the performance criteria is given in Section 2. The first step is to evaluate the performance criterion for a given control configuration — we consider co-located sensor/controls only. The finite element model is described in Section 3, and the continuum model in Section 4. The dynamic state space model is seen to be identical in both cases except for state space dimension. Section 5 derives the performance criteria valid for either model, except that for the distributed parameter model we obtain explicit closed form solutions. Section 6 compares the numerical results obtained by the two models, showing complete agreement. As a byproduct of our analysis, we obtain a solution to the problem of optimal placement of actuators to minimize mean square attitude error — in Section 7. An optimization problem *per se* — a canonical one — is treated in Section 8 and by virtue of our explicit formulas for performance indices in terms of structure/control parameter, shown to be "trivial" and even capable of analytical solution — in contrast to computer solutions using FEM or truncated modal models as in [9, 10].

We should note that structural engineers (Noor, *et al.* [1, 2] and Renton [3]) have already voiced the advantages of continuum models in preliminary structure design — what is new here is the application to control design, to the Controls-Structures Interaction problem.

1. The Physical Article

The physical structure (Figure 1) is a lattice of rectangular bays, each single-laced single-bay. Offset at each end is an antenna. The controllers are force and moment actuators with co-located attitude as well as rate sensors stationed at arbitrary locations along the structure. Table 1 is a breakdown of the parameters describing each bay.

TABLE 1
Element Properties

	Battens	Longitudinal Bars	Diagonal Bars	Cross Bracing in Battens
Length L	b	ℓ	d	δ
Sectional Area A	A_b	A_ℓ	A_d	A_δ
Elastic Modulus E	E_b	E_ℓ	E_d	E_δ
Mass Density m	ρ_b	ρ_ℓ	ρ_d	ρ_δ
Element Mass = ρAL	m_b	m_ℓ	m_d	m_δ
Element Stiffness = EA/L	S_b	S_ℓ	S_d	S_δ

The beam geometry is shown in Figure 2. By the "nominal" structure, we shall mean the following choice of structural parameters:

$$b = \ell\,; \qquad d = \delta = \sqrt{2}\,\ell$$

$$A_b = A_\ell = A_d = A_\delta = A$$

$$= E_b = E_\ell = E_d = E_\delta = E$$

$$= \rho_b = \rho_\ell = \rho_d = \rho_\delta = \rho$$

$$L = n\ell\,;$$

Nominal values:

$$n = 9$$

$$\ell = 3 \text{ m}$$

$$\rho = 3250$$

$$E = 2.759 \times 10^{11}$$

$$A = 2.468 \times 10^{-4} \text{m}^2 .$$

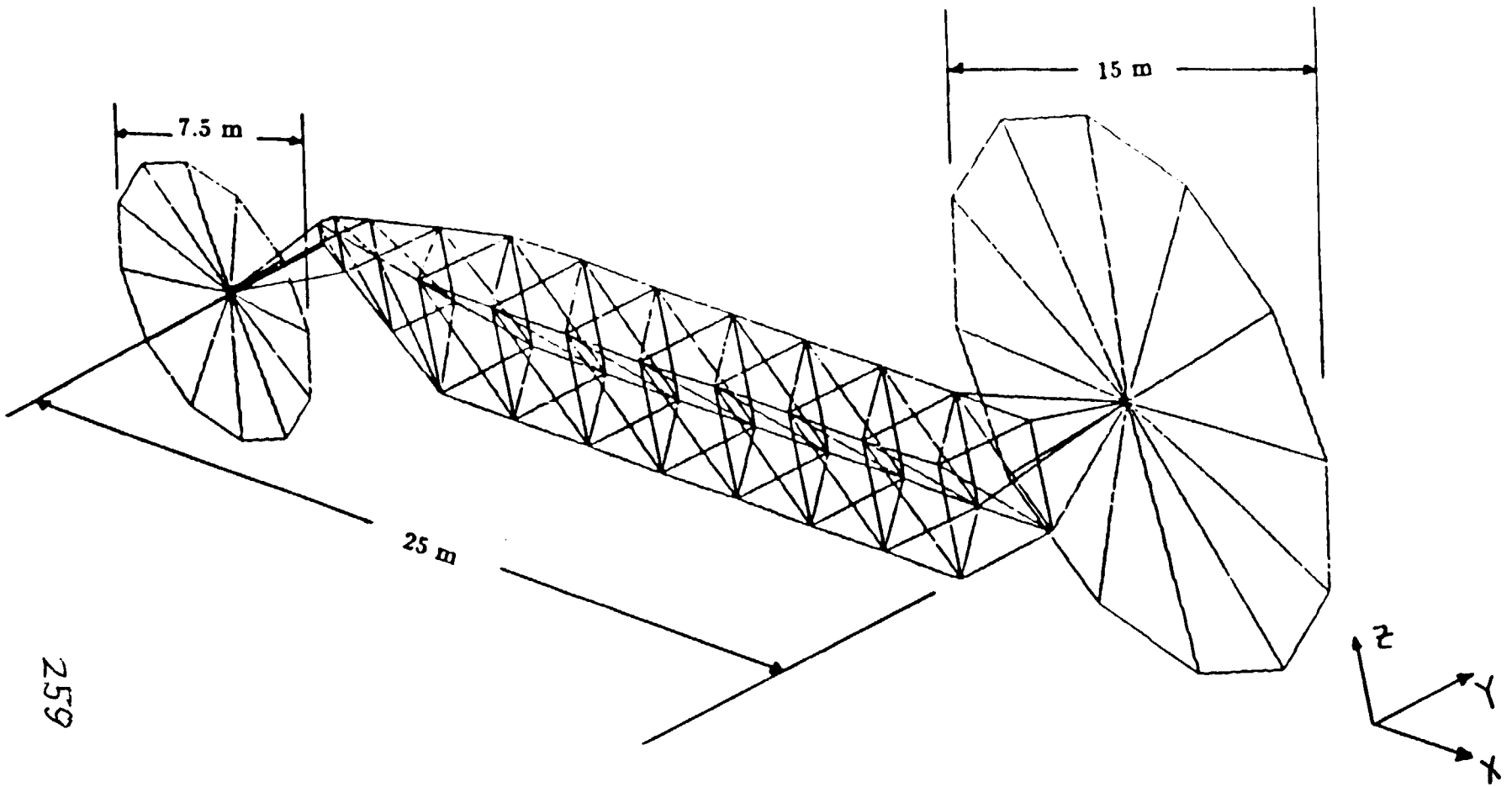


Figure 1. Earth Pointing Satellite (EPS)

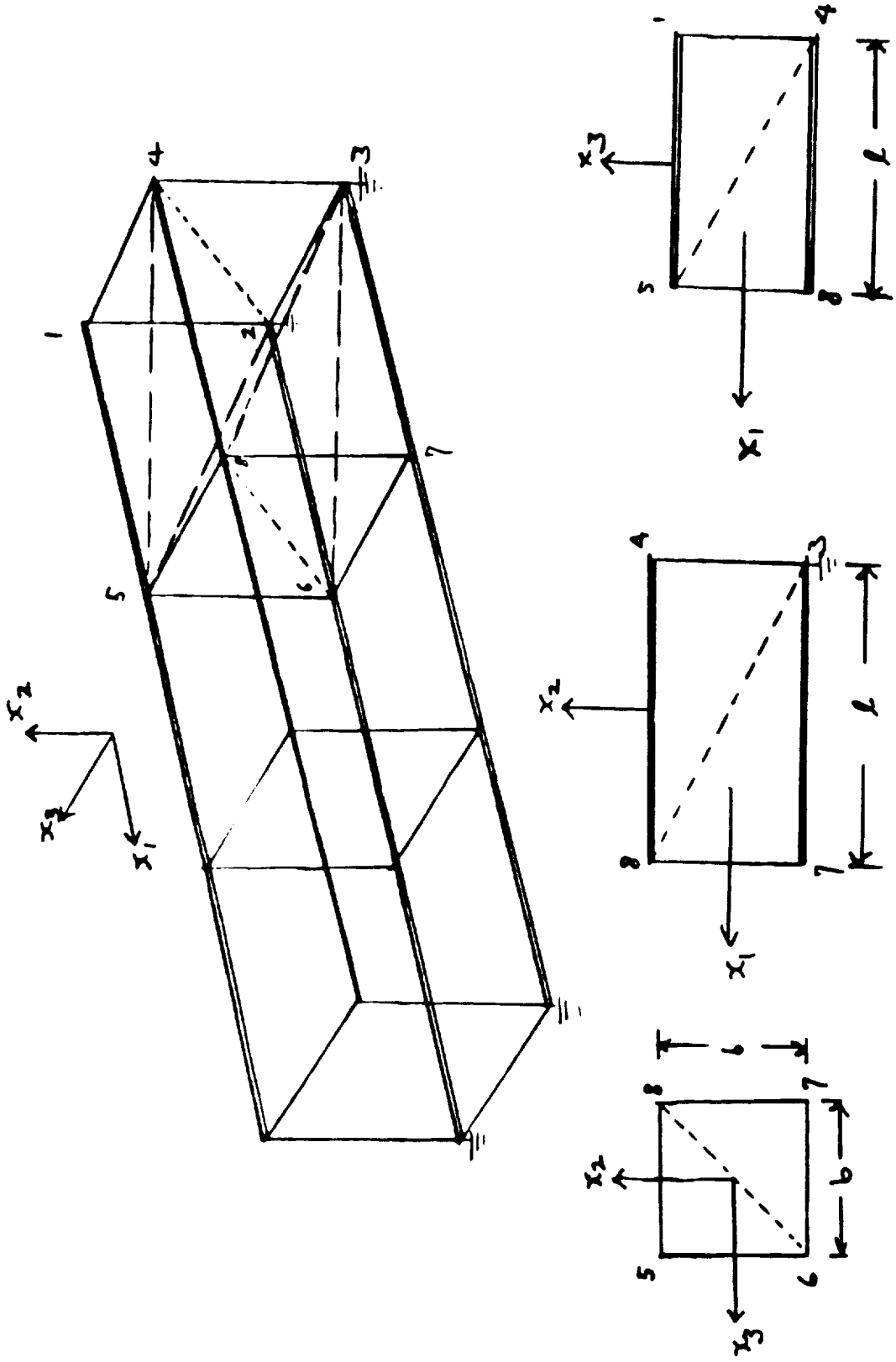


Figure 2. Beam Geometry

2. Performance Criteria

As in previous CIS optimization studies (see [9] and the references therein) the objective is to minimize the total mass of the structure, including the controller mass, subject to meeting specified performance requirements, which we shall now describe in mathematical terms.

The performance criteria chosen are:

(i) the mean-square attitude error due to sensor noise (using co-located sensors/ actuators). As we shall see, this actually depends on the steady state "tip" response to step inputs so that "noise" notions can be eschewed if necessary.

(ii) the "stability margin": defined as the sum of the absolute values of the real parts of the closed-loop eigenvalues. This is one measure of stability among very many (see, e.g., [10]). We choose this one because it is essentially equivalent to any other one but has the advantage that we can derive a simple closed-form expression for it.

Let us now define the criteria more precisely. First of all we assume the control law to be PD ("proportional plus derivative") as in classical servo design. Let $v_p(t)$ denote the "displacement" or "attitude" vectors at the sensor locations and let $v_r(t)$ the rate. Then $U(t)$ the control is defined to be

$$U(t) = \alpha v_p(t) + \gamma v_r(t) \tag{2.1}$$

where α and γ are (positive) scalar "gains." This is not altogether a "simplifying" assumption — that the scalar rate feedback is actually optimal is shown in [4]. The beam-axis being the x_1-axis, we have, with L denoting the beam-length:

$$0 \leq x_1 \leq L .$$

Let $f(0), f(L)$ denote the 6-DOF displacement vectors at the ends. Let

$$f(0) = \begin{vmatrix} u(0) \\ v(0) \\ w(0) \\ \phi_1(0) \\ \phi_2(0) \\ \phi_3(0) \end{vmatrix} ; \quad f(L) = \begin{vmatrix} u(L) \\ v(L) \\ w(L) \\ \phi_1(L) \\ \phi_2(L) \\ \phi_3(L) \end{vmatrix} \tag{2.2}$$

where u, v, w are the 3-DOF displacements, ϕ_1 the torsion angle about the beam axis, and ϕ_2, ϕ_3 about the mutually perpendicular axes. Then the mean square attitude error is defined by

$$\sigma_a^2 = \overline{u(0)^2 + u(L)^2} + \overline{v(0)^2 + v(L)^2} + \overline{w(0)^2 + w(L)^2} + |r_0|^2 \overline{\phi_1(0)^2} + |r_L|^2 \overline{\phi_1(L)^2} , \tag{2.3}$$

where the bars denote time averages, $|r|$ being the length of the moment arm as required.

Under our feedback law (often referred to as "positive-definite" feedback) the closed-loop system is guaranteed to be stable. Assuming no damping in th structure (as we shall indeed do), the real parts of the closed-loop eigenvalues are guaranteed negative (see Section 5) — or if we assume the structure is already damped we have stability enhance ment corresponding to the increment, the real parts being now more negative. Let σ_i denote the real part increment corresponding to the ith closed-loop mode. Then the infinite series

$$\sum_i^\infty -\sigma_i \tag{2.4}$$

converges. Denote the sum by σ_s. We shall take this to be the "stability index" — the higher the index, the more stable.

3. The Finite Element Model

Since most of the techniques in developing the FEM are standard, we shall only present the relevant numerical data. Each bar is taken as a truss element with 6 DOF. There are $(13+5)$ elements per bay, and hence $(13\times 9 + 5) = 122$ elements for 9 bays. There are 40 nodes with 3 DOF each, so that the stiffness matrix A and the mass/inertia matrix M are 120×120. The state vector is thus 120×1. The displacements along the axis of the truss are then expressed (or, rather, extrapolated):

$$u(k\ell) = u(k\ell, 0, 0) = \frac{u(k\ell,0-b/2) + u(k\ell,0+b/2) + u(k\ell,-b/2,0) + u(k\ell,+b/2,0)}{4}$$

$$v(k\ell) = v(k\ell, 0, 0) = \frac{v(k\ell,0-b/2) + v(k\ell,0+b/2) + v(k\ell,-b/2,0) + v(k\ell,+b/2,0)}{4}$$

$$w(k\ell) = w(k\ell, 0, 0) = \frac{w(k\ell,0-b/2) + w(k\ell,0+b/2) + w(k\ell,-b/2,0) + w(k\ell,+b/2,0)}{4}$$

$$\phi_1(k\ell) = \frac{1}{2}\left[\frac{w(k\ell,+b/2,0) - w(k\ell,-b/2,0)}{b} - \frac{v(k\ell,0,+b/2) - v(k\ell,0,-b/2)}{b}\right]$$

$$\phi_2(k\ell) = \frac{u(k\ell,0,+b/2) - u(k\ell,0,-b/2)}{b}$$

$$\phi_3(k\ell) = \frac{u(k\ell,+b/2,0) - u(k\ell,-b/2,0)}{b}$$

where k is an integer and $0 \le k\ell \le L$. Allowing for m controllers at $k = k_1, k_2, \ldots, k_m$, the corresponding relations can be represened by a $6m\times 120$ matrix acting on the state vector. (We consider in this paper $m = 1$, 2 or 3.) Let B denote the transpose of this matrix. Then the state space dynamics with co-located sensors can be described by:

$$M\ddot{x} + Ax + BU + BN_a = 0 \tag{3.1}$$

with sensor data:

$$v_p(t) = B^*x(t) + N_p(t) \tag{3.2}$$

$$v_r(t) = B^*x(t) + N_r(t) \tag{3.3}$$

where $U(\cdot)$ denotes the control; $N_a(\cdot)$, $N_p(\cdot)$, $N_r(\cdot)$ model additive noise taken as (mutually independent) white Gaussian with spectral density matrices d_aI, d_pI, d_rI respectively, I being the identity matrix.

4. The Continuum Model

As we noted in Section 1, the problem of producing an "exact," "three-D" continuum model for a real-world structure like the truss we are considering can be a formidable one — although research in this area looks promising [5]. One way out of this difficulty is to exploit where possible the special nature of the truss — in our case it is a lattice of bays along the same axis numerous enough so that it is even visually "beam-like." In that case there are many ways to approximate by "one-D" beams — without going into the details of this theory, suffice it to say that the approach by Noor and Russell [2] is the one adapted here. We thus create an "equivalent" (referring to [2] for the precise sense) one-dimensional *anisotropic* Timoshenko beam as follows.

u denoting axial (longitudinal) displacement (along the x_1-axis);
ϕ_1 the torsion angle about this axis ,

w, ϕ_2 denoting the transverse bending displacement in the x_1-x_3 plane and torsion angle about the x_2-axis ,

v, ϕ_3 denoting the transverse bending displacement in the x_1-x_2 plane and torsion angle about the x_3-axis ,

the three axes being mutually perpendicular; $0 \le x_1 \le L$,
L being the beam length = $n\ell$; n = number of bays

The Timoshenko equations (valid between control nodes) are:

$$m_{11}\ddot{u} - c_{11}u'' - c_{14}v'' - c_{15}w'' - c_{15}\phi_2' + c_{14}\phi_3' = 0$$

$$m_{22}\ddot{v} - c_{44}v'' - c_{14}u'' + c_{44}\phi_3' = 0$$

$$m_{33}\ddot{w} - c_{55}w'' - c_{15}u'' - c_{55}\phi_2' = 0$$

$$m_{44}\ddot{\phi}_1 - c_{66}\phi_1'' - c_{36}\phi_2'' - c_{26}\phi_3'' = 0$$

$$m_{55}\ddot{\phi}_2 + m_{56}\ddot{\phi}_3 + c_{15}u' + c_{55}w' - c_{36}\phi_1'' + c_{55}\phi_2 - c_{33}\phi_2'' - c_{23}\phi_3'' = 0$$

$$m_{66}\ddot{\phi}_3 + m_{56}\ddot{\phi}_2 - c_{14}u' - c_{44}v' - c_{26}\phi_1'' - c_{23}\phi_2'' + c_{44}\phi_3 - c_{22}\phi_3'' = 0$$

where the superdots denote time-derivatives and the primes, space derivatives. The coeffi-

cients of these dynamic equations are related to the truss parameters as follows: (cf. [2]): The mass coefficients are given by:

$$m_{11} = m_{22} = m_{33} = \frac{4m_b + 4m_\ell + 4m_d + m_\delta}{\ell}$$

$$m_{44} = 2m_{55} = 2m_{66} = \frac{\ell(8m_b + 12m_\ell + 8m_d + m_\delta)}{6\mu^2}$$

$$m_{56} = \frac{\ell m_\delta}{12\mu^2}$$

The stiffness (flexibility) c_{ij} are given by:

$$c_{11} = 4\ell S_\ell + \frac{4\ell S_b S_d \mu^2}{S_d + S_b(\ell + \mu^2)}$$

$$c_{44} = \frac{c_{14}}{\mu} = c_{55} = \frac{c_{15}}{\mu} = \frac{2\ell S_b S_d}{S_d + S_b(\ell + \mu^2)}$$

$$c_{22} = c_{33} = \frac{\ell^3 S_\ell}{\mu^2} + \frac{\ell^3 S_b S_d}{4(S_d + S_b(\ell + \mu^2))}$$

$$c_{23} = \frac{\ell^3 S_b S_d}{4(S_d + S_b(\ell + \mu^2))}$$

$$c_{66} = -2c_{26} = -2c_{36} = \frac{\ell^3 S_b S_d}{\mu^2(S_d + S_b(\ell + \mu^2))}$$

where

$$\mu = \frac{\ell}{b} .$$

In order not to complicate matters unduly in this demonstration, we shall freeze all parameters except the cross-sectional area A which will then be the structural parameter to be optimized. In this case

$$c_{11} = \frac{(40 + 24\sqrt{2})EA}{9 + 4\sqrt{2}} \quad \text{(Newton)}$$

$$c_{14} = -c_{15} = c_{44} = c_{55} = \frac{2EA}{1 + 2\sqrt{2}} \quad \text{(Newton)}$$

$$c_{22} = c_{33} = \frac{(2725 + 1476\sqrt{2})EA\ell^2}{2628 + 1336\sqrt{2}} \quad \text{(Newton)m}^2$$

$$c_{23} = -\left(\frac{(97 + 140\sqrt{2})EA\ell^2}{2628 + 1336\sqrt{2}}\right) \qquad \text{(Newton)m}^2$$

$$c_{26} = -c_{36} = \frac{1}{2}c_{66} = \frac{(16 + 33\sqrt{2})EA\ell^2}{296 + 130\sqrt{2}} \qquad \text{(Newton)m}^2$$

$$m_{11} = m_{22} = m_{33} = (8 + 5\sqrt{2})A\rho \qquad \text{kg/m}$$

$$m_{44} = 2m_{55} = 2m_{66} = \frac{(20 + 9\sqrt{2})A\ell^2\rho}{6} \qquad \text{kg} \cdot \text{m}$$

$$m_{56} = -\frac{(A\ell^2\rho)}{6\sqrt{2}} \qquad \text{kg} \cdot \text{m}\,.$$

Once the coefficients c_{ij}, m_{ij} are defined (on whatever basis), we can develop the generic state space dynamic model analogous to the FEM formulas (3.1), (3.2), (3.3):

$$M\ddot{x} + Ax + Bu + BN_a = 0 \tag{4.1}$$

$$\left.\begin{aligned} v_p(t) &= B^*x(t) + N_p(t) \\ v_r(t) &= B^*\dot{x}(t) + N_a(t) \end{aligned}\right\} \tag{4.2}$$

where $N_a(\cdot)$, $N_p(\cdot)$, $N_r(\cdot)$ are white Gaussian noise with spectral density d_aI, d_pI, d_rI respectively.† Only, the dimension of the state $x(t)$ is not finite. The technique of derivation is also different, in particular in the role of the energy. See [6, 7] for details. Here we can only summarize the basic results.

Case 1: One Controller

We begin with constructing the state space model for one controller ("midcontroller"), and offset masses at each end, referring to [4, 6] again for more details and to [7] where the general case of distributed control is treated. Thus the state $x(t)$ at time t is defined by

$$x(t) = \begin{vmatrix} f(\cdot, t) \\ f(0, t) \\ f(s_2, t) \\ f(L, t) \end{vmatrix}$$

† See [7] for generalization to arbitrary diagonal matrices.

where s parametrizes the beam axis, zero denoting one end and L the other, L being the total beam length, and s_2 denotes the location of the mid-controllers and $f(\cdot, t)$ denotes a (6×1 vector) function of s, $0 \le s \le L$, representing displacements and angles:

$$f(s, t) = \begin{vmatrix} u(s,t) \\ v(s,t) \\ w(s,t) \\ \phi_1(s,t) \\ \phi_2(s,t) \\ \phi_3(s,t) \end{vmatrix} .$$

The stiffness operator A is defined as follows:

$$Ax = \begin{vmatrix} g \\ v \end{vmatrix} ; \qquad x = \begin{vmatrix} f(\cdot) \\ f(0) \\ f(s) \\ f(L) \end{vmatrix}$$

$$g(s) = -A_2 f''(s) + A_1 f'(s) + A_0 f(s) , \qquad 0 < s < s_2 \; ; \; s_2 < s < L .$$

The derivative $f'(\cdot)$ has a discontinuity at $s = s_2$, and

$$v = \begin{vmatrix} -L_1 f(0) - A_2 f(0) \\ -A_2(f'(s_2+) - f'(s_2-)) \\ L_1 f(L) + A_2 f'(L) \end{vmatrix}$$

and thus defined, the potential energy

$$= [Ax, x] = \int_0^L \left[H \begin{vmatrix} f'(s) \\ f(s) \end{vmatrix} , \begin{vmatrix} f'(s) \\ f(s) \end{vmatrix} \right] ds \ge 0$$

where

$$H = \begin{vmatrix} C_1 & 0 & 0 & -C_2 \\ 0 & C_3 & 0 & 0 \\ 0 & 0 & & \\ -C_2^* & 0 & & A_0 \end{vmatrix}$$

$$A_2 = \begin{vmatrix} C_1 & 0 \\ 0 & C_3 \end{vmatrix}$$

$$C_1 = \begin{vmatrix} c_{11} & c_{14} & c_{15} \\ c_{14} & c_{44} & 0 \\ c_{15} & 0 & c_{55} \end{vmatrix}$$

$$C_3 = \begin{vmatrix} c_{66} & c_{36} & c_{26} \\ c_{36} & c_{33} & c_{23} \\ c_{26} & c_{23} & c_{22} \end{vmatrix}$$

$$A_1 = \begin{vmatrix} 0 & C_2 \\ -C_2^* & 0 \end{vmatrix}$$

$$C_2 = \begin{vmatrix} 0 & -c_{15} & c_{14} \\ 0 & 0 & c_{44} \\ 0 & -c_{55} & 0 \end{vmatrix}$$

$$A_0 = \text{Diag. } [\, 0,\ 0,\ 0,\ 0,\ c_{55},\ c_{44} \,]$$

$$L_1 = \begin{vmatrix} 0 & -C_2 \\ 0 & 0 \end{vmatrix}.$$

B is defined by

$$BU = \begin{vmatrix} 0 \\ 0 \\ U \\ 0 \end{vmatrix}, \qquad U \text{ is } 6\times 1$$

and

$$B^*x = f(s_2) \qquad (6\times 1).$$

The mass/inertia operator M is defined

$$Mx = \begin{vmatrix} M_0 f(\cdot) \\ M_{b,0} f(0) \\ M_c f(s_2) \\ M_{b,L} f(L) \end{vmatrix}$$

where

$$M_0 = \begin{vmatrix} m_{11} & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & m_{22} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & m_{33} & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & m_{44} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & m_{55} & m_{56} \\ \cdot & \cdot & \cdot & \cdot & m_{56} & m_{66} \end{vmatrix}$$

$$\text{Diag. } M_{b_0} = (m_0 , \ m_0 , \ m_0 , \ \text{Diag. } I_0)$$

where m_0 is the offset mass at $s = 0$ and I_0 is its moment of inertia about zero, and similarly

$$\text{Diag. } M_{b_L} = (m_L , \ m_L , \ m_L , \ \text{Diag. } I_L) .$$

See also [6] for more on M_{b_0}, M_{b_L}.

$$M_c = \begin{vmatrix} m_c & 0 & 0 & 0 & 0 & 0 \\ 0 & m_c & 0 & 0 & 0 & 0 \\ 0 & 0 & m_c & 0 & 0 & 0 \\ 0 & 0 & 0 & & & \\ 0 & 0 & 0 & & I_c & \\ 0 & 0 & 0 & & & \end{vmatrix}$$

where m_c is the force actuator (rotor) mass and I_c the moment of inertia of the moment actuator about its center of gravity. The m.s. attitude error matrix is defined by

$$\lim_{T \to \infty} \left\{ \frac{1}{T} \int_0^T f(0, t) f(0,)^* \, dt + \frac{1}{T} \int_0^T f(L, t) f(L, t)^* \, dt \right\}$$

and σ_a^2 is the sum of the diagonal terms as defined.

Case 2: Two Controllers

Next we consider the case of two controllers, one at each end. Here, since there is no mid-controller, we may delete that entry in the state. Thus

$$x(t) = \begin{vmatrix} f(\cdot, t) \\ f(0, t) \\ f(L, t) \end{vmatrix}$$

$$Mx = \begin{vmatrix} M_0 f(\cdot, t) \\ M_{b,0} f(0, t) \\ M_{b,L} f(L, t) \end{vmatrix} .$$

where the end-masses m_0, m_L must now include the actuator moving masses, and similarly for the moment of inertia matrices. We shall use the notation

$$M_c = \begin{vmatrix} M_{b,0} & 0 \\ 0 & M_{b,L} \end{vmatrix}$$

With $U(\cdot)$ denoting the control vector, (12×1), we have

$$BU = \begin{vmatrix} 0 \\ U \end{vmatrix}$$

$$B^*x = \begin{vmatrix} f(0, t) \\ f(L, t) \end{vmatrix} .$$

Finally

$$Ax = \begin{vmatrix} g \\ \mathrm{v} \end{vmatrix}$$

$$\mathrm{v} = \begin{vmatrix} -L_1 f(0) - A_2 f'(0) \\ L_1 f(L) + A_2 f'(L) \end{vmatrix}$$

$$g(s) = -A_2 f''(s) + A_1 f'(s) + A_0 f(s) , \qquad 0 < s < L .$$

Here the mean square attitude error-matrix

$$= \lim_{T\to\infty} \frac{1}{T} \int_0^T (B^*x(t))(B^*x(t))^* \, dt .$$

Case 3: Three Controllers

In this case we have a mid-controller at $s = s_2$ as well as a controller at each end. Here

$$x(t) = \begin{vmatrix} f(\cdot, t) \\ f(0, t) \\ f(s_2, t) \\ f(L, t) \end{vmatrix} ,$$

in other words the same state vector as in Case 1.

$$Mx(t) = \begin{vmatrix} M_0 f(\cdot, t) \\ M_{b,0} f(0, t) \\ M_{b,2} f(s_2, t) \\ M_{b,L} f(L, t) \end{vmatrix},$$

where

$$M_{0,2} = \begin{vmatrix} m_c & 0 & 0 & 0 & 0 & 0 \\ 0 & m_c & 0 & 0 & 0 & 0 \\ 0 & 0 & m_c & 0 & 0 & 0 \\ 0 & 0 & 0 & & & \\ 0 & 0 & 0 & & I_c & \\ 0 & 0 & 0 & & & \end{vmatrix}$$

where m_c is the force actuator moving mass and I_c is the moment of inertia about its center of gravity, corresponding to the "mid-controller."

$$BU = \begin{vmatrix} 0 \\ U \end{vmatrix}$$

$$B^*x = \begin{vmatrix} f(0, t) \\ f(s_2, t) \\ f(L, t) \end{vmatrix}.$$

We can calculate σ_a^2 from the diagonal terms of the mean square error matrix:

$$\text{Diag.} \lim_{T \to \infty} \frac{1}{T} \int_0^T (B^*x(t))(B^*x(t))^* \, dt .$$

which now has 18 entries. Once again we adopt the notation:

$$M_c = \begin{vmatrix} M_0 & \cdot & \cdot & \cdot \\ \cdot & M_{b,0} & \cdot & \cdot \\ \cdot & \cdot & M_{b,2} & \cdot \\ \cdot & \cdot & \cdot & M_{b,L} \end{vmatrix}.$$

5. Formulas for Performance Criteria

We shall now develop formulas for the Performance Critera. First the mean square attitude error: Using either model FEM (3.1)-(3.3) or Continuum (4.1)-(4.2) we have the state space form:

$$M\ddot{x} + Ax + BU + BN_a = 0 \tag{5.1}$$

$$v_p = B^*x + N_p \tag{5.2}$$

$$v_r = B^*\dot{x} + N_r \tag{5.3}$$

and

$$U(t) = \alpha v_p(t) + \gamma v_r(t) . \tag{5.4}$$

Substituting this control law into the state equations we have:

$$M\ddot{x} + (A + \alpha BB^*)x + \gamma BB^*\dot{x} + B(N_a + \alpha N_p + \gamma N_r) = 0 . \tag{5.5}$$

The steady-state output covariance matrix

$$R_a = E((B^*x(t))(B^*x(t))^*) = \left(\frac{\alpha^2 d_p + d_a + \gamma^2 d_r}{2\gamma}\right)(B^*(A + \alpha BB^*)^{-1}B) \tag{5.6}$$

where the matrix part:

$$B^*(A + \alpha BB^*)^{-1}B$$

is recognized as the steady-state input-output response matrix: it is the value at $\omega = 0$ of the input-output transfer-function:

$$B^*((i\omega)^2M + A + \alpha BB^* + (i\omega)\gamma BB^*)^{-1}B .$$

The scalar factor

$$d = \frac{\alpha^2 d_p + d_a + \gamma^2 d_r}{2\gamma} \tag{5.7}$$

consolidates the "noise" part. For the case of two controllers the mean square attitude error σ_a^2 can be calculated from

$$\text{Diag. } R_a .$$

For the third case of three controllers it is given by

$$\text{Diag. } LR_a L^*$$

where

$$LU = \begin{vmatrix} f(0) \\ f(L) \end{vmatrix} .$$

For Case 1 with one controller only, we have to calculate

$$E(f(0, t)\, f(0, t)^*)$$

$$E(f(L, t)\, f(L, t)^*) .$$

Separately, expressing each as a transformation of the state:

$$L_1 x = f(0)$$

$$L_2 x = f(L) .$$

In the FEM version, we have thus to invert the matrix

$$(A + \alpha BB^*)$$

which in our case is 120×120 — and of course can be done only numerically. For the continuum model however we have to invent the operator

$$(A + \alpha BB^*)^{-1}$$

but — and this is the main point of departure — this can be done *analytically*. Referring to [7] for details, here we shall simply enumerate the formulas below.

Case 1: One Controller Only

$$E[f(0, t)\, f(0, t)^*] = \text{Diag.} \left[\frac{1}{\alpha}, \frac{1+s_2^2}{\alpha}, \frac{1+s_2^2}{\alpha}, \frac{1}{\alpha}, \frac{1}{\alpha}, \frac{1}{\alpha}\right] d$$

$$E[f(L, t)\, f(L, t)^*] = \text{Diag.} \left[\frac{1}{\alpha}, \frac{1 + (L-s_2)^2}{\alpha}, \frac{1 + (L-s_2)^2}{\alpha}, \frac{1}{\alpha}, \frac{1}{\alpha}, \frac{1}{\alpha}\right] d$$

(each is a diagonal matrix!). Steady state step response corresponding to step input U:

$$f(0,\infty) = \frac{1}{\alpha}\begin{vmatrix} I_3 & \begin{matrix}0 & 0 & 0\\ 0 & 0 & -s_2\\ 0 & s_2 & 0\end{matrix} \\ 0 & I_3 \end{vmatrix} U$$

$$f(L,\infty) = \frac{1}{\alpha}\begin{vmatrix} I_3 & \begin{matrix}0 & 0 & 0\\ 0 & 0 & L-s_2\\ 0 & s_2-L & 0\end{matrix} \\ 0 & I_3 \end{vmatrix} U$$

Note: Controller at $s = s_2$.

Case 2: Two End Controllers

$$\text{Diag. } E[f(0,t)\,f(0,t)^*] = \text{Diag. } E[f(L,t)\,f(L,t)^*] .$$

And the first four diagonal terms in either matrix are given in order by:

$$= \left[1 + \frac{\mu\beta}{\lambda\beta-2}\right]\left(\frac{1}{2\alpha}\right)d$$

$$= \left[1 + \frac{\mu(\lambda\beta-1)}{\beta(\lambda\beta-2)}\right]\left(\frac{1}{2\alpha}\right)d$$

$$= \left[1 + \frac{\mu(\lambda\beta-1)}{\beta(\lambda\beta-2)}\right]\left(\frac{1}{2\alpha}\right)d$$

$$= \left[\frac{1}{2\alpha} + \frac{L}{4}\frac{c_{33} + \frac{\alpha L}{2} + \frac{c_{66}}{4}}{\left(c_{33} + \frac{\alpha L}{2} + \frac{c_{66}}{4}\right)\left(c_{66} + \frac{\alpha L}{2}\right) - \frac{c_{66}^2}{2}}\right]d$$

where

$$\mu = \frac{\alpha L}{2c\delta} + \frac{L^2}{4}$$

$$\beta = \mu + 1$$

$$\lambda = \mu + 2 + \frac{(c_{11} - 2c)(2\alpha + c\delta L)}{2\alpha c\delta}$$

$$c = c_{44}$$

$$\delta = \frac{1}{1 + \frac{L^2\gamma}{12}}$$

$$\gamma = \frac{c_{44}}{c_{33} - \frac{c_{66}}{4}} .$$

Case 3: Three Controllers

Here we have to express the answers in terms of 6×6 matrices. Thus let

$$D_{11} = \frac{E[f(0,\ t)\ f(0,\ t)^*]}{d}$$

$$D_{22} = \frac{E[f(s_2,\ t)\ f(s_2,\ t)^*]}{d}$$

$$D_{33} = \frac{E[f(L,\ t)\ f(L,\ t)^*]}{d} .$$

Then

$$D_{11} = d_{11}(s_2) - d_{12}(s_2)(d_{22}(s_2) + (d_{11}(L-s_2)^{-1} - \alpha)^{-1})^{-1} d_{12}^*(s_2)$$

$$D_{22} = (d_{22}^{-1}(s_2) + d_{11}(L-s_2)^{-1} - \alpha)^{-1}$$

$$D_{33} = (\alpha I + m_3(L-s_2))^{-1}\{\alpha I + m_3(L-s_2) + m_{21}(L-s_2)(D_{22})m_{21}^*(L-s_2)\} \times (\alpha I + m_3(L-s_2))^{-1}$$

where

$$d_{11}(s) = ((\alpha + m_1(s)) - m_{21}(s)^*(\alpha + m_3(s))^{-1} m_{21}(s))^{-1}$$

$$d_{22}(s) = ((\alpha + m_3(s)) - m_{21}(s)(\alpha + m_1(s))^{-1} m_{21}^*(s))^{-1}$$

$$d_{12}(s) = -d_{11}(s)m_{21}(s)^*(\alpha I + m_3(s))^{-1}$$

$$m_1(s) = \begin{vmatrix} \frac{c\delta(s)}{s} A_{11} & \frac{c\delta(s)}{2} A_{12} \\ \frac{c\delta(s)}{2} A_{12}^* & \frac{C_3}{s} + \frac{cs\delta(s)}{4} A_{22} \end{vmatrix}$$

$$m_3(s) = \begin{vmatrix} \frac{c\delta(s)}{s} A_{11} & -\frac{c\delta(s)}{2} A_{12} \\ -\frac{c\delta(s)}{2} A_{12}^* & \frac{C_3}{s} + \frac{cs\delta(s)}{4} A_{22} \end{vmatrix}$$

$$m_{21}(s) = \begin{vmatrix} -\frac{c\delta(s)}{s}A_{11} & -\frac{c\delta(s)}{2}A_{12} \\ \frac{c\delta(s)}{2}A_{12}^{*} & -\frac{C_3}{s} + \frac{cs\delta(s)}{4}A_{22} \end{vmatrix}$$

where

$$\delta(s) = \frac{1}{1 + \frac{\gamma s^2}{12}}$$

$$A_{11} = \begin{vmatrix} \frac{1}{\delta(s)}\left(\frac{c_{11}}{c} - 2(1-\delta(s)\right) & 1 & -1 \\ 1 & 1 & 0 \\ -1 & 0 & 1 \end{vmatrix}$$

$$A_{12} = \begin{vmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{vmatrix}$$

$$A_{22} = \begin{vmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

Stability Margin Formula

From (5.5) it is clear that the closed-loop eigenfunctions are specified by

$$\lambda^2 M\phi + (A + \alpha BB^*)\phi + \gamma\lambda BB^*\phi = 0$$

(again, irrespective of whether we are using the FEM version (3.1)–(3.3) or the continuum version (4.1), (4.2)) where ϕ denotes the eigenvector ("mode shape"), with the eigenvalue λ specified by

$$\lambda^2[M\phi, \phi] + [(A + \alpha BB^*)\phi, \phi] + \lambda\gamma[B^*\phi, B^*\phi] = 0$$

where we may normalize the mode shape ϕ so that $[M\phi, \phi] = 1$. Since $\alpha \geq 0$ and $\gamma \geq 0$,

it follows that the eigenvalue λ is given by:

$$\lambda = -\sigma \pm i\omega$$

where

$$\sigma = -\gamma \frac{\|B^*\phi\|^2}{2}$$

$$\omega^2 = \omega_0^2 - \sigma^2$$

$$\omega_0^2 = [(A + \alpha BB^*)\phi, \ \phi] \ .$$

Thus the closed-loop eigenvalues have strictly negative real parts.

It is shown in [8] that the sum of the absolute values of the closed-loop eigenvalues is given by

$$\gamma \ \text{Tr.} \ M_c^{-1} \tag{5.8}$$

in all cases, where in M_c only moving parts of the actuators must be considered (as opposed to the stationary mass such as the armature mass). The simplicity of this formula is striking when compared with taking the sum of the inverses of the absolute values of the real parts of the closed loop eigenvalues for a finite number of modes as in [10]. We may note that for zero natural damping (or with damping, if we consider only the increment),

$$\sum_1^\infty \frac{1}{|\sigma_i|} = \infty$$

for any continuum model. Again, (5.8) applies for the FEM version (3.1)-(3.3), as well as for the continuum model (4.1), (4.2).

6. Numerical Results

In this section we compare the numerical results by the two methods for the mean square attitude errors (and equivalently, tip response), for each of the three cases: one, two and three controllers — using the nominal values (cf. Section 1) for the truss parameters.

Case 1: One Controller

FEM

The tip response $f(0, \infty)$ and $f(L, \infty)$ was calculated for:

$$\alpha = c_{44} = 35571851.2 ; \qquad s_2 = 0, \frac{L}{9}, \frac{L}{3}, \frac{L}{2},$$

$$\alpha = 10{,}000 ; \qquad s_2 = 0, \frac{L}{9}, \frac{L}{3}, \frac{L}{2},$$

$$\alpha = 100 ; \qquad s_2 = 0, \frac{L}{9}, \frac{L}{3}, \frac{L}{2},$$

both $f(0, \infty)$ and $f(L, \infty)$ values were exactly the same as the values predicted by the continuum model and hence are not displayed.

Case 2: Two Controllers

$$b_{ii} = \text{Diag. } B^*(A + \alpha I)^{-1} B , \qquad i = 1, ..., 12$$

		FEM	Continuum
$\alpha = 1$	$b_{11} = b_{77}$	0.4998	0.5
	$b_{22} = b_{33} = b_{88} = b_{99}$	0.99727	0.99727
	$b_{44} = b_{10,10}$	0.50028	0.5
$\alpha = 50$	$b_{11} = b_{77}$	0.01	0.01
	$b_{22} = b_{33} = b_{88} = b_{99}$	0.019945	0.019945
	$b_{44} = b_{10,10}$	0.01	0.01
$\alpha = c_{44} = 3.557 \times 10^7$	$b_{11} = b_{77}$	2.2978×10^{-8}	2.2978×10^{-8}
	$b_{22} = b_{33} = b_{88} = b_{99}$	2.80545×10^{-8}	2.80546×10^{-8}
	$b_{44} = b_{10,10}$	2.478×10^{-8}	2.478×10^{-8}

Case 3: Three Controllers

$$\alpha = 3.557 \times 10^7$$

$$b_{ii} = i = 1, \ldots, 18$$

Actuator Position	$b_{11} \times 10^8$ FEM	$b_{11} \times 10^8$ Continuum
$s = \frac{L}{9}$	1.5119	1.5127
$s = \frac{3L}{9}$	1.8366	1.8370
$s = \frac{4L}{9}$	1.9461	1.9465
$s = \frac{5L}{9}$	2.0330	2.0333
$s = \frac{6L}{9}$	2.1031	2.1033
$s = \frac{7L}{9}$	2.1602	2.1603

In other words the FEM and the continuum gave exactly the same numerical results within (the SUN-386i) computer accuracy in all cases.

SUM OF M.S. AXIAL DISPLACEMENTS AT BOTH ENDS

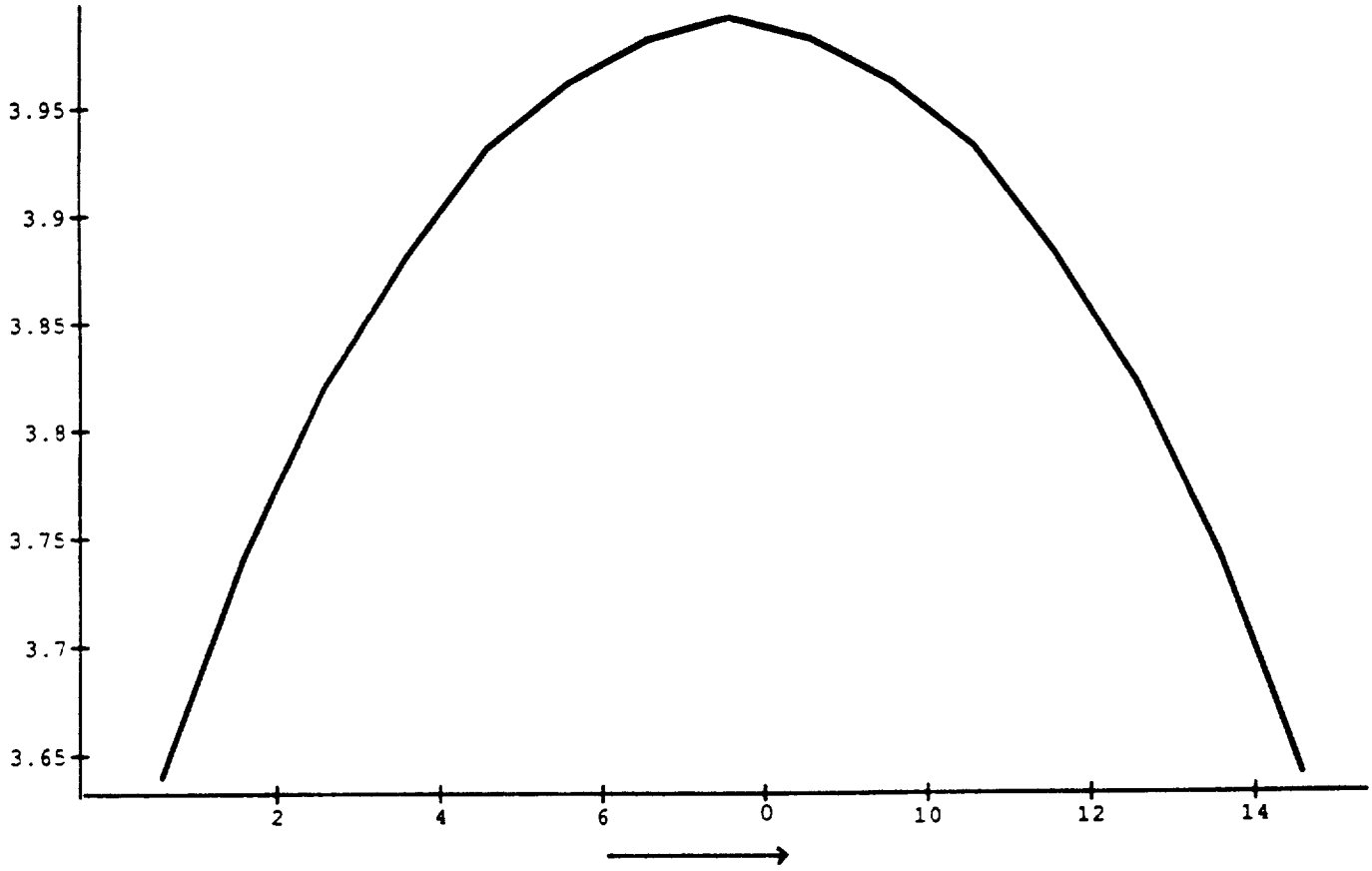


FIGURE 3

MIDACTUATOR POSITION: INCREMENTS L/16

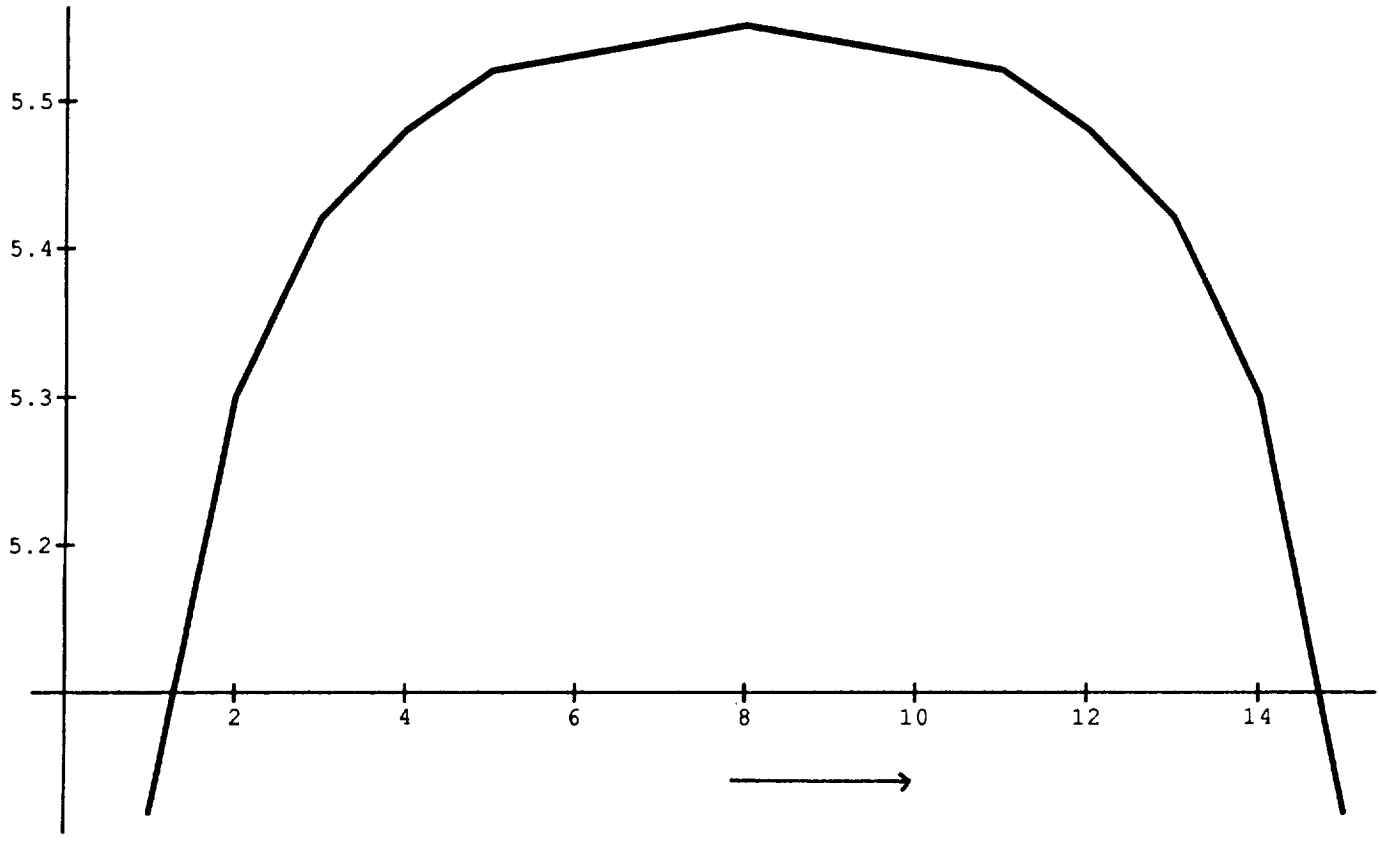


FIGURE 4

MIDACTUATOR POSITION: INCREMENTS L/16

SUM OF M.S. TORSION AT BOTH ENDS

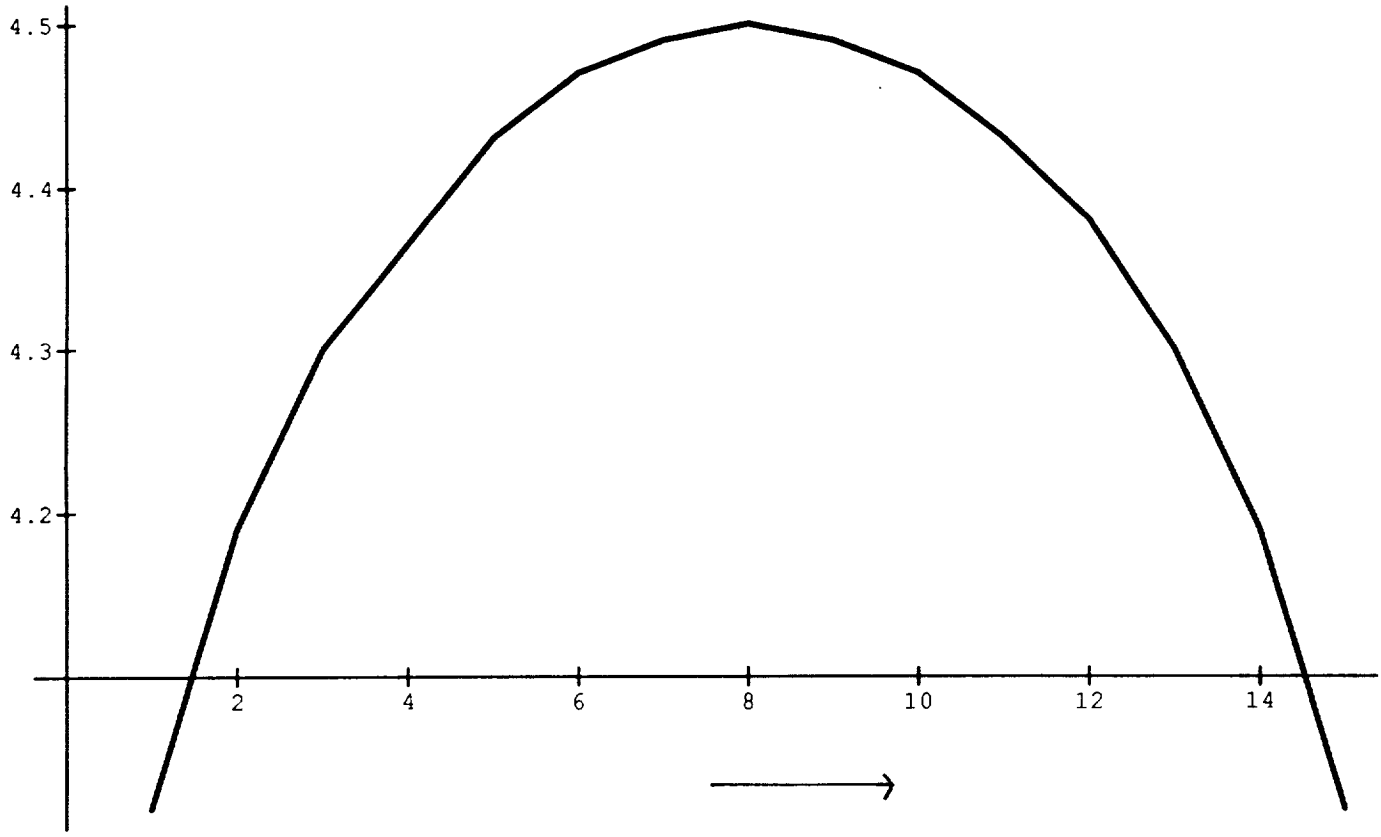


FIGURE 5

MIDACTUATOR POSITION: INCREMENTS L/16

M.S. AXIAL DISPLACEMENT AT ACTUATOR LOCATION

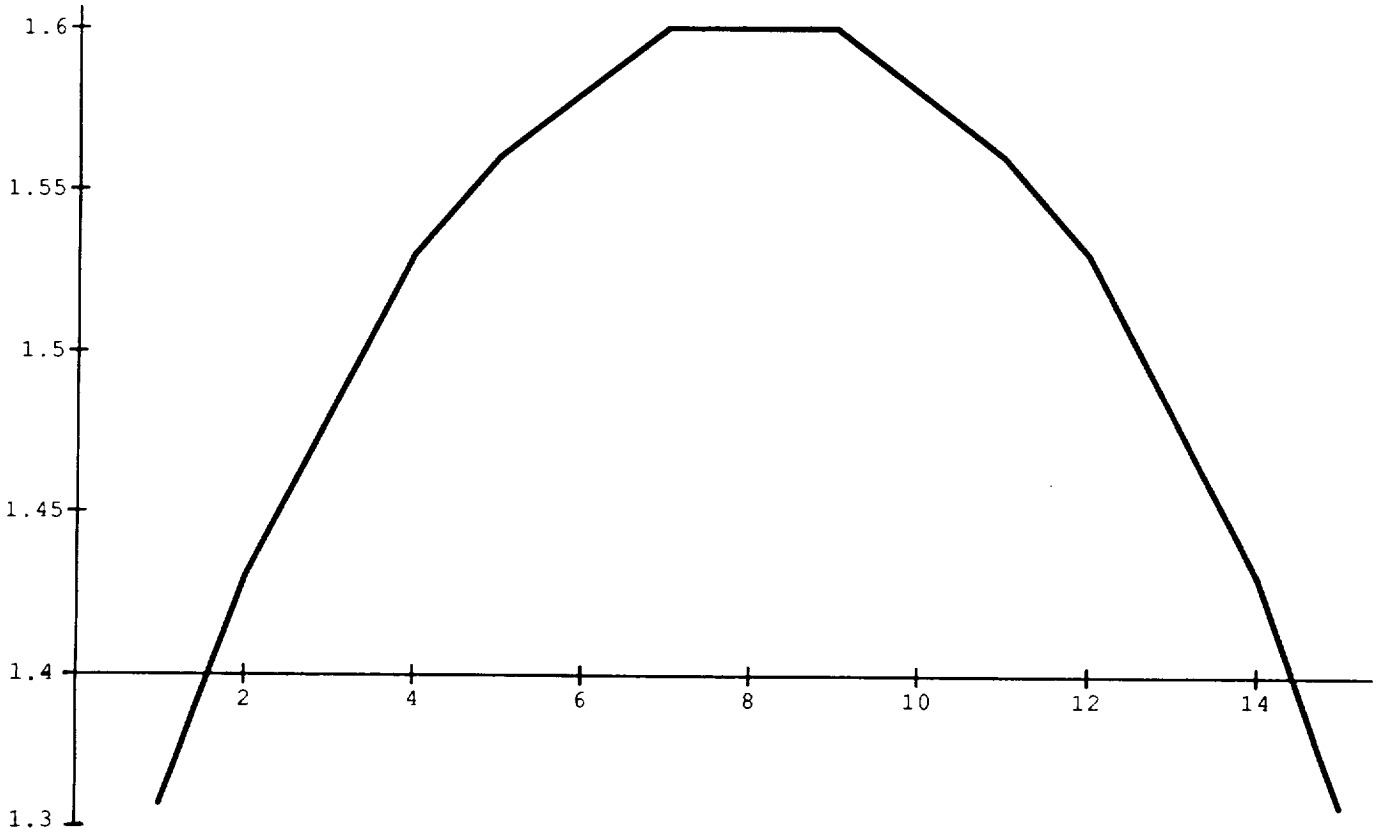


FIGURE 6

MIDACTUATOR POSITION: INCREMENTS L/16

7. Optimal Location of Controllers

As a byproduct of our theory, we can examine the problem of the optimal actuators placement that minimizes the mean square attitude for any given choice of control gains and structure parameters.

Case 1: One Actuator

Here we take the criterion as the sum of the mean square displacements at both ends. We calculate explicitly that

$$\frac{[E[f(0,t)\ f(0,t)^*] + E[f(L,t)\ f(L,t)^*]]}{d}$$

$$= \text{Diag.}\left[\frac{2}{\alpha}, \frac{2 + s^2 + (L-s)^2}{\alpha}, \frac{2 + s^2 + (L-s)^2}{\alpha}, \frac{2}{\alpha}, \frac{2}{\alpha}, \frac{2}{\alpha}\right]$$

from which it is clear that the optimal placement is in the middle:

$$s = \frac{L}{2}.$$

Only the bending-displacement is affected by actuator position.

For Case 2 of two actuators there is no placement problem since one is required to be at each end.

Case 3: Three Actuators

Here we can consider the optimal placement of the midactuator while the other two are fixed one at each end. The behavior of the sum of the mean square errors (setting $\alpha = 1$) at both ends for the axial mode is shown in Figure 3; and the bending mode is shown in Figure 4; and the axial torsion in Figure 5. In all cases we see that the worst position is at the middle! The best place is at either end. Finally Figure 6 shows the mean square axial displacement at the actuator location. Again the worst place is the middle.

8. Optimization

We shall now treat a canonical optimization problem currently studied by FEM and truncated modal models [9, 10]. The objective is to minimize the total mass — structure and control — subject to meeting specified indices of performance. Here we take them to be:

(a) mean square attitude error due to sensor noise less than or equal to fixed value

(b) stability index: sum of the absolute values of the real parts of the closed loop eigenvalues to be not less than a fixed value.

We shall see that the problem can be solved analytically by virtue of the formulas we have developed using continuum models.

The structural parameter we shall use is the cross-sectional area A of the longerons (assumed to be the same for battens and cross-bars). Other parameters being fixed, the structure mass is then proportional to A. (The extension to the case of nonequal areas only complicates the algebra, as can be seen from the expressions (cf. Section 3) for the flexibility coefficients.) The control mass has to be subdivided into a stationary mass (armature mass, for example) and a moving mass (rotor mass, for example) since only the latter is involved in the stability index formula. The stationary mass is of course related to the moving mass — for simplicity we shall take it to be inversely proportional to the rotor mass. The control parameters are the attitude and rate gains α and γ. These of course will need to be constrained not to exceed prescribed limits. Thus we have the following formulation (nominal values for all structure parameters except A):

$$\text{Structure mass} = N\ell\rho A$$

$$\text{Control stationary mass} = \frac{k}{m}$$

$$\text{Moving mass} = m$$

$$\text{Total mass} = N\ell\rho A + \frac{k}{m} + m$$

$$\text{Stability index} = \frac{\gamma}{M + m}$$

(where M denotes the contribution of the end-masses). For the truss considered,

$$N = (76 + 46\sqrt{2}) .$$

Finally the mean square attitude error — to be specific, we shall consider the case of two controllers, one at each end; and take the sum of the mean square displacements at either end. First we express these in terms of the structural parameter A — we have thus to use the expressions we have derived for the flexibility coefficients $\{c_{ij}\}$ in terms of A in Section 5 and substitute them into the formulas for mean square errors for two controllers in Table 3 under Case 2. In doing so we shall also take advantage of the simplification possible by noting that for the nominal value of $L = 27$ meters, we can readily calculate that

$$\lambda\beta \geq 2$$

so that we may replace both $(\lambda\beta - 1)$ and $(\lambda\beta - 2)$ by $\lambda\beta$. Thus the first four diagonal terms in

$$2 \text{ Diag. } E[f(0, t) f(0, t)^*]$$

$$(= 2 \text{ Diag. } E[f(L, t) f(L, t)^*])$$

are given in order by:

$$f(\alpha, \gamma, A) = \frac{1}{\alpha}\left[1 + \frac{\frac{L^2}{4} + \frac{L}{2c\delta}\frac{\alpha}{EA}}{2 + \frac{L^2}{4} + \frac{L}{2c\delta}\frac{\alpha}{EA} + \left(\frac{c_{11} - 2c}{2c\delta}\right)\left(\frac{2\alpha + Lc\delta EA}{\alpha}\right)}\right] d$$

$$\frac{1}{\alpha}\left[1 + \frac{\frac{L^2}{4} + \frac{L}{2c\delta}\frac{\alpha}{EA}}{1 + \frac{L^2}{4} + \frac{L}{2c\delta}\frac{\alpha}{EA}}\right] d$$

$$\frac{1}{\alpha}\left[1 + \frac{\frac{L^2}{4} + \frac{L}{2c\delta}\frac{\alpha}{EA}}{1 + \frac{L^2}{4} + \frac{L}{2c\delta}\frac{\alpha}{EA}}\right] d$$

$$\left[\frac{1}{\alpha} + \frac{L}{2EA}\frac{c_{33} + \frac{c_{66}}{4} + \frac{L}{2}\frac{\alpha}{EA}}{\left(c_{33} + \frac{c_{66}}{4} + \frac{L}{2}\frac{\alpha}{EA}\right)\left(c_{66} + \frac{L}{2}\frac{\alpha}{EA}\right) - \frac{c_{66}^2}{2}}\right] d$$

where now

$$c_{11} = \frac{40 + 24\sqrt{2}}{9 + 4\sqrt{2}}$$

$$c = \frac{2}{1 + 2\sqrt{2}}$$

$$c_{33} = n\left(\frac{2725 + 1476\sqrt{2}}{2628 + 1336\sqrt{2}}\right)$$

$$c_{66} = n\left(\frac{32 + 66\sqrt{2}}{296 + 130\sqrt{2}}\right)$$

$$n = \text{number of bays} ; \qquad L = n\ell$$

$$\delta = \frac{1}{1 + \frac{L^2}{12} \cdot \frac{c}{(c_{33} - c_{66}/4)}} .$$

with d as given by (5.7). These formulas enable us to draw conclusions concerning the dependence on the cross-sectional area A without resorting to numerical computer calculations. We see that all the errors decrease as A increases. The axial error decreases from $\frac{2d}{\alpha}$ at $A = 0$ to $\frac{d}{\alpha}$ at $A = \infty$; similarly the torsion error. The bending error is least affected, decreasing from $\frac{d}{\alpha}$ at $A = 0$ to $\left(\frac{L^2}{4 + L^2}\right)\frac{d}{\alpha}$ at $A = \infty$. In all cases the minimal mean square error is at most 3 db less than the maximum!

For the optimization let us fix on the mean square bending error as being the easiest analytically: let

$$f(\alpha, \gamma, A) = \frac{d}{\alpha}\left[1 - \frac{1}{1 + \frac{L^2}{4} + \frac{L}{2c\delta}\frac{\alpha}{EA}}\right] .$$

Thus the optimization problem is that of minimizing

$$N\ell\rho A + \frac{k}{m} + m$$

subject to:

$$\frac{\gamma}{M + m} \geq \sigma_s^2$$

$$f(\alpha, \gamma, A) \leq \sigma_a^2 .$$

The first inequality can clearly be reversed to read

$$\frac{M + m}{\gamma} \leq \frac{1}{\sigma_s^2} .$$

We note that the objective functional

$$N\ell\rho A + \frac{k}{m} + m$$

is infinitely smooth and trivially convex, and the constraints are also infinitely smooth and convex. Hence we are assured of the existence of a minimum (which is further verified to be unique). Moreover we can go to the Langrange parameter formulation and minimize:

$$\left(N\ell\rho A + \frac{k}{m} + m\right) + \lambda_1 \left(\frac{M + m}{\gamma}\right) + \lambda_2 f(\alpha, \gamma, A)$$

where λ_1, $\lambda_2 \geq 0$ are the Lagrange parameters. See [11] for the standard results that are applicable here.

Compared to the FEM versions [9, 10] this is a "trivial" problem and complete "analytical" solution is possible. We omit the details since our primary aim in this paper is to demonstrate the simplicity of the optimization problem relative to the FEM versions.

Acknowledgement

Grateful acknowledgement is made of several informal discussions with E. Armstrong and S. Joshi of the CSI Analytical Design Methods Team at NASA Langley FRC on the problem; and of the FEM calculations by H-C. Wang.

References

1. A. K. Noor and W. C. Russell. "Anisotropic Continuum Models for Beamlike Lattice Trusses." *Computer Methods in Applied Mechanics and Engineering*, 1986, pp. 257-277.

2. A. K. Noor and C. M. Andersen. "Analysis of Beamlike Trusses." *Computer Methods in Applied Mechanics and Engineering*, 1978, pp. 53-70.

3. J. D. Renton. "The Beam-like Behavior of Space Trusses." *AIAA Journal*, Vol. 22, No. 2 (Feb 1984).

4. A. V. Balakrishnan. "Feedback Control for Stabilization of Flexible Multibody Structures." In: *Differential Equations: Stability and Control.* New York: Marcel Dekker, 1990.

5. L. W. Taylor. "Distributed Parameter Modeling for the Control of Flexible Spacecraft." In: *Proceedings of the NASA-UCLA Workshop on Computational Techniques in Identification and Control of Flexible Flight Structures, Lake Arrowhead, November 1989.* New York-Los Angeles: OSI Publications, 1990. Pp. 87-114.

6. A. V. Balakrishnan. "Combined Structures-Controls-Integrated Optimization Using Distributed Parameter Models." In: *Proceedings of the NASA-UCLA Workshop on Computational Techniques in Identification and Control of Flexible Flight Structures, Lake Arrowhead, November 1989.* OSI Publications, 1990. Pp. 70-86.

7. A. V. Balakrishnan. "Dynamics of Anisotropic Timoshenko Beams with Distributed Controls." To appear.

8. A. V. Balakrishnan. "A Mathematical Formulation of the SCOLE Control Problem, Part II: Optimal Compensator Design." NASA CR 181720. December 1988.

9. M. Salama, J. Garha, L. Demsetz and F. Udwadia. "Simultaneous Optimization of Control Structures." *Computational Mechanics,*" Vol. 3 (1988), pp. 257-282.

10. Sandeep Gupta and S. M. Joshi. "An Integrated Control/Structure Design Method Using Multiobjective Optimization." Presented at the 4th NASA Workshop on Computational Control of Flexible Aerospace Systems, Williamsburg, Virginia, July 1990. Preprint.

11. B. T. Polyak. *Introduction to Optimization.* OSI Publications, 1987.

end file: july90

Presented at *4th NASA Workshop on Computational Control of Flexible Aerospace Systems*, July 11-13, 1990, Williamsburg, VA.

Control and Dynamics of a Flexible Spacecraft during Stationkeeping Maneuvers

D. Liu and J. Yocum
Hughes Aircraft Company

D.S. Kang
C. S. Draper Laboratory

Summary

A case study of a spacecraft having flexible solar arrays is presented. A stationkeeping attitude control mode employing both earth and rate gyro reference signals and a flexible vehicle dynamics modelling and implementation is discussed.

The control system is designed to achieve both pointing accuracy and structural mode stability during stationkeeping maneuvers. Reduction of structural mode interactions over the entire mode duration is presented. The control mode employing a discrete-time observer structure is described to show the convergence of the spacecraft attitude transients during ΔV thrusting maneuvers without pre-loading thrusting bias to the on-board control processor. The simulation performance using the three-axis, body-stabilized nonlinear dynamics is provided.

The details of a five-body nonlinear dynamics model are discussed. The spacecraft is modelled as a central rigid body having cantilevered flexible antennas, a pair of flexible articulated solar arrays, and two gimballed momentum wheels. The vehicle is free to undergo unrestricted rotations and translations relative to inertial space. A direct implementation of the equations of motion will be compared to an indirect implementation that uses a symbolic manipulation software to generate rigid body equations. A generalization of this approach to this class of flexible vehicles will be provided.

1. Introduction

Three-axis body stabilized spacecrafts having solar wings with significant structural flexibility may exhibit rigid-flex coupling effects during a typical stationkeeping maneuver. One of the primary concerns for the design of three-axis stabilized spacecraft is the structural mode interaction with the attitude control system. In addition, the dynamic analysis and the control performance evaluation are sensitive to the rigid-flex modelling accuracy. This paper presents a case study on the design, analysis and digital simulation of a microprocessor-based stationkeeping control system of a 3-panel communication spacecraft using thrusters as control actuators. It discusses the control system to achieve

modal stabilization, the dynamic model development and validation, and the technique for closed-loop digital simulation.

The vehicle under discussion is a three-axis stabilized geosynchronous communication satellite. The vehicle is powered by photovoltaic solar arrays and are controlled in its on-orbit operation by a combination of momentum wheels, reaction control thrusters, and solar array motion. This satellite is modelled as a central rigid body having cantilevered flexible antennas, a pair of flexible articulated solar arrays, and two gimballed momentum wheels. The vehicle is free to undergo unrestricted rotations and translations relative to inertial space. The solar arrays rotate relative to the central body in response to the action of control torques. The momentum wheels are assumed to be controlled such that their motions relative to the central body are prescribed. In the deployed configuration, the solar arrays contain 75% of the total inertia of the satellite with only 7% of the total mass. Each solar array has 4 cantilevered frequencies below 1 Hz.

The solar wing flexibility is fully coupled into the body roll and yaw dynamics because the flexible solar wings are fixed about the roll and yaw axes of the central body, while articulating about its pitch axis. The pitch coupling depends upon the nonlinearity of the solar wing drive and its friction characteristics. The control bandwidth of the on-orbit normal mode is usually designed at a frequency well below the first structural mode so that the solar wing flexibility does not interact seriously with the normal mode controller. However, a relative high control bandwidth is needed to maintain pointing accuracy in the presence of a large thrusting disturbance. The disturbance torques are primarily induced by the offset of spacecraft center of gravity (CG) from the geometric or pressure center of maneuvering jets as well as the thrust mismatch. The sensed spacecraft flexible dynamics interact with the stationkeeping controllers, which may result in structural mode instability at high loop gains.

.During the thrusting maneuver, the dominant modes coupling in the stationkeeping control are phase stabilized using the lead inherent in the sensed gyro rates together with the phase-lead notch filters, while the non-dominant modes at higher frequencies are gain-stabilized. After the thrusting maneuver, any residual rates must be nulled by an order of magnitude in preparation for a smooth transfer back to the normal control mode. This paper will focus on the stationkeeping control during the thrusting maneuver only. The thrusting bias about each control axis is estimated such that a fast convergence on the bias estimates can be achieved without an open-loop torque pre-bias by ground command, although this feature is also included in the design. The control loop bandwidths are designed as high as possible to meet pointing requirements, while still achieving adequate modal stability. Simulation results demonstrate stationkeeping control performance from a typical 2-jet (5 lbf each) south maneuver under the worst case maneuvering conditions are presented.

To verify the accuracy of the dynamics model, two approaches are taken. In the first approach, the direct approach, the equations of motion for the vehicle are derived from fundamental momentum principles. The flexible appendages are modelled with conventional lumped mass model employing stiffness matrices to characterize the internal energy. Transformations to appendage modal coordinates are made and a reduction in the number of elastic degrees of freedom is achieved through their truncation. Simplifying assumptions are made regarding the magnitude of certain nonlinear kinematic terms based on operational considerations. The final set of governing equations are coded in a first order form suitable for numerical integration.

In the second approach, the indirect approach, an unconventional method is employed. The "rigid" portion of the equations and the code is obtained from a symbolic

manipulation software. The "rigid/flex" coupling terms are derived and implemented. The details of this approach will be presented later in this paper. The two approaches were numerically compared through a set of chosen open loop comparison tests.

2. Performance Requirements and Control System Descriptions

Performance Requirements

The spacecraft under study requires at least 400 bi-weekly south maneuvers with a maximum duration of 120 sec per maneuver using two 5 lbf thrusters. East/west corrections are 5 sec short burn each. Factors affecting pointing accuracy during the stationkeeping maneuver are earth sensor noise, rate gyro noise, gyro rate bias estimation errors, thrusters pulse-to-pulse repeatability, spacecraft CG offset, CG migration due to propellant motion, flexibility of solar wings and reflectors, thrust mismatch, thruster misalignment, on-time/off-time thruster delay, thruster plume-impingement, etc. Effects due to environment disturbance such as solar torques, magnetic torques, wing torques, etc., are assumed to be negligible. The goal is to maintain body transients to within ± 0.1 deg in roll/pitch and ± 0.2 deg in yaw. The control loops should stabilize structural mode oscillation seen on spacecraft attitude and provide stability range in the presence of structural mode frequency uncertainty.

Control Algorithms

The stationkeeping control is executed through special control algorithms that run when the spacecraft is in Stationkeeping Mode. Due to the spacecraft CG offset and variation in thrust pulse amplitude, various thrusting disturbance torques about the control axes may be induced when the maneuver thrusters are activated. Stationkeeping Mode provides thruster control for a maneuver execution and autonomous attitude control to limit body transient errors and maintain pointing accuracy during maneuvers. Attitude control in roll and pitch axes is achieved using earth referenced signals and rate integrating gyro data with 5 lbf thrusters. A rate integrating gyro is used as rate references in yaw control.

Figure 2.1 shows the functional block diagram of the 3-axis stationkeeping attitude control system during maneuvers. The control logic for each of roll and pitch rate loops in Stationkeeping Mode is combined with the gyro referenced attitude estimator for position, rate and acceleration bias estimates and the proportional controller for control acceleration command generation. Using earth referenced pitch and roll position signals, effects due to gyro drift can be minimized by an on-board software calibration to the raw rate measurements. The yaw control logic in Stationkeeping Mode is identical to the roll and pitch channels except for the yaw gyro bias estimate, which is constant based on premaneuver calibration estimate. The technique for rate gyro calibration will not be discussed in this paper.

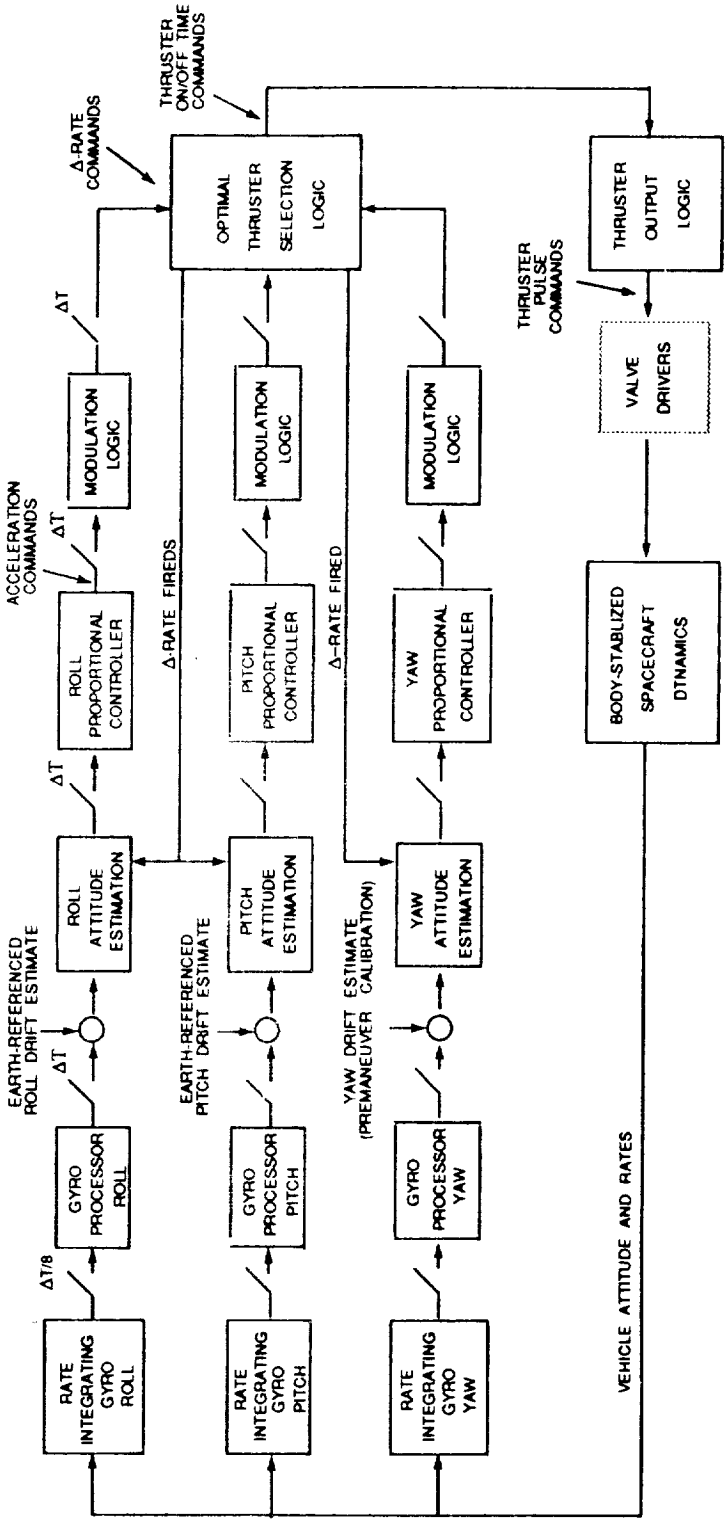


Figure 2.1 Functional Control Block Diagram of Stationkeeping Mode

The earth referenced pitch and roll position errors are sensed at 2.0345 Hz, i.e. every 30 real time interrupts (RTIs) at 16.384 msec per RTI, to supply position references for the roll and pitch gyro calibration. The angle estimate is obtained by adding the gyro bias estimate to the raw rate measurement, and integrating the resulting rate. For the yaw control loop, it does not perform the yaw gyro calibration during maneuver, since no yaw position position references are available in Stationkeeping Mode. Instead, yaw gyro bias is estimated in the normal mode prior to maneuver, and its value is held throughout the maneuver.

The gyro referenced rate measurements with calibration are sensed every 8 RTIs to supply rate references to a third-order attitude estimator for each axis. Each attitude estimator performs two functions. First, it integrates the gyro rate (after correction for bias) to obtain a position estimate. Both the roll and pitch gyro calibration along with integrations operate during the premaneuver gyro calibration period as well as throughout the maneuver. Second, it estimates the spacecraft angular rate and acceleration bias about the respective axis. The roll and yaw attitude estimators also include the effect of roll-yaw coupling due to spinning wheel momentum.

A proportional controller is employed for each axis to determine the control acceleration commands based upon the position, rate and acceleration bias estimates. The control acceleration commands are held constant over each control sample period. The on-board optimal thruster selection (OTS) logic selects available thrusters and determines necessary thruster on/off command duration to valve drivers based upon the minimum fuel consumption. The selected 5 lbf thrusters are turned on/off for commanded durations to deliver the control momentum equivalent to the commands, and achieve attitude corrections during maneuvers. The detailed technique for conversion of the control acceleration commands to thruster commands will not be discussed further. Although the control sample period is designed to be commandable, it must be selected to meet the needs of the control processor thruput and avoid structural mode instabilities as well.

3. Flexible Spacecraft Model Descriptions

The mechanical idealization of the satellite is illustrated in Figure 3.1. The model consists of a central body, which is considered to be rigid and to which are mounted a set of reaction control thrusters. Cantilevered to the central body are a complement of structurally flexible antennas. Two distinct, structurally flexible solar arrays are hinge connected to the central body. The arrays can rotate independently about parallel drive axes in response to control torques, which are assumed to be known functions of time. Two independent, identical, rigid axisymmetric, variable-speed momentum wheels are mounted to the central body through two-axis gimbal mechanisms. The complete motions of the wheels relative to the central body are assumed to be prescribed functions of time.

The communications antennas of the actual satellite are capable of limited articulation relative to the central body. However, because these rotations are small in both magnitude and rate, their influence on the vehicle's overall attitude dynamics was deemed negligible, and these degrees of freedom were not included in the model. While the solar arrays will be virtually identical under nominal circumstances, they are treated as structurally distinct to accommodate more general conditions.

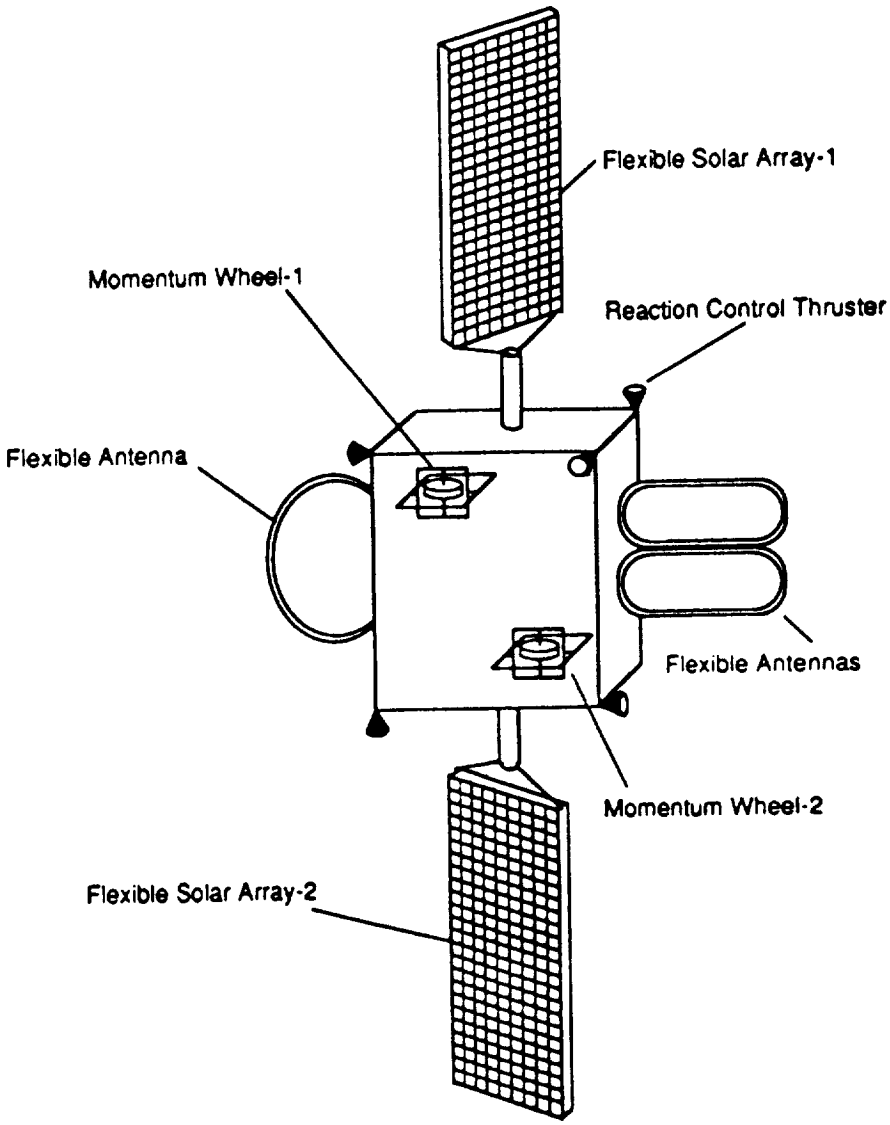


Figure 3.1. Mechanical Idealization of a satellite.

The deformations of the antenna and solar array structures are assumed to be linear elastic in character and small in magnitude. The respective appendages are modelled as collections of point masses interconnected by massless elastic structure. Stiffness matrices are used to define the elastic restoring forces acting internal to these assemblies. Ultimately, modal coordinate transformations are introduced for each appendage and the final motion equations are cast in terms of truncated sets of those variables.

Cantilevered frequencies of an individual solar array and of an assembled antennas are provided in Table 3.1.

Table 3.1. Characteristic Cantilevered Frequencies of the Appendages.

Mode #	Solar Array Frequency (Hz)	Antennas Frequency (Hz)
1	.118	1.586
2	.355	1.792
3	.705	1.953
4	.835	2.043
5	1.825	4.235
6	2.725	4.867
7	3.167	5.323
8	4.914	5.893
9	6.161	13.366
10	6.905	19.558

4. Control Design and Analysis

Linearized Spacecraft Open Loop Dynamics

Let $\{b\}$ be the spacecraft body frame, $\{s\}_w$ be the solar wing frame and C_w be the direction cosine matrix at the wing-to-body angle α such that $\{s\}_w = C_w\{b\}$, where w = n (North wing) or s (South wing). Figure 4.1 is a geometry showing the spacecraft with the thrusting forces and torques and the disturbance created by thruster plume impingement; in which we assume that point i is the pressure center on North wing where the resultant plume force vector applies.

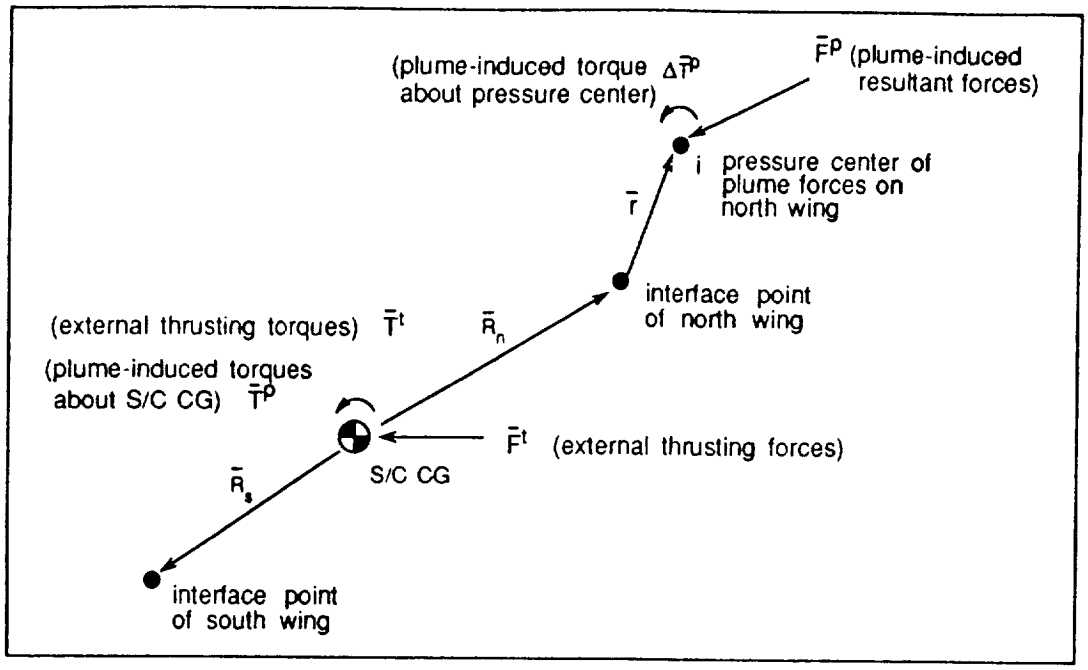


Figure 4.1 A Geometry showing External Thrusting Forces/Torques and Plume Disturbance

The linearized body-stabilized spacecraft hybrid dynamics with articulated solar wings are given in Equations (4.1) - (4.6).

$$m\ddot{x} + P_n\ddot{\eta}_n + P_s\ddot{\eta}_s = F^t + F^p \qquad \text{(S/C translational motion)} \qquad (4.1)$$

$$I_{s/c}\dot{\omega} + I_n e_2 \dot{\omega}_n + I_s e_2 \dot{\omega}_s + Q_n\ddot{\eta}_n + Q_s\ddot{\eta}_s = T^t + T^p \qquad \text{(S/C rotational motion)} \qquad (4.2)$$

$$e_2^T I_n \dot{\omega} + I_2^n \dot{\omega}_n + e_2^T Q_n^e \ddot{\eta}_n = T_n^{swd} \qquad \text{(North wing pitch dynamics)} \qquad (4.3)$$

$$e_2^T I_s \dot{\omega} + I_2^s \dot{\omega}_s + e_2^T Q_s^e \ddot{\eta}_s = T_s^{swd} \qquad \text{(South wing pitch dynamics)} \qquad (4.4)$$

$$P_n^T \ddot{x} + Q_n^T \dot{\omega} + (e_2^T Q_n^o)^T \dot{\omega}_n + \ddot{\eta}_n + 2\zeta\Lambda \dot{\eta}_n + \Lambda^2 \eta_n = \Phi_i^T \begin{bmatrix} C_n F^p \\ \Delta T_i^p \end{bmatrix}$$ (North wing flex dyn.) (4.5)

$$P_s^T \ddot{x} + Q_s^T \dot{\omega} + (e_2^T Q_s^o)^T \dot{\omega}_s + \ddot{\eta}_s + 2\zeta\Lambda \dot{\eta}_s + \Lambda^2 \eta_s = 0$$ (South wing flex dyn.) (4.6)

where

m	=	total S/C mass
$I_{s/c}$	=	S/C mass inertias
x	=	S/C translational position vector
ω	=	S/C angular rate vector
I_n, I_s	=	pitch inertias of wings about their interface points
ω_n, ω_s	=	relative pitch angular rates of wings
η_n, η_s	=	modal variables of wings
P_n^o, P_s^o	=	rigid-flex translational coupling matrices of wings about their interface points
P_n, P_s	=	rigid-flex translational coupling matrices of wings about the S/C CG
Q_n^o, Q_s^o	=	rigid-flex rotational coupling matrices of wings about their interface points
Q_n, Q_s	=	rigid-flex rotational coupling matrices of wings about the S/C CG
T_n^{swd}, T_s^{swd}	=	solar wing torques
$F^p, \Delta T^p$	=	plume induced force and torque about pressure center
Φ	=	mode shape at point i of North wing (a nx6 matrix with 3 translational and 3 rotational deformation, where n is the number of modes in concern)
Λ	=	cantilever mode frequencies for each wing
ζ	=	structural damping factor
e_2	=	$[0\ 1\ 0]^T$

The model above is with articulated solar wings driven by the wing torques about their hinge axes. To fully include the flexibility of wings, the dynamic inertias of each wing referenced to its interface point must contain at least 99% of the roll or yaw moment of inertia about the same point, or, the pitch inertia excluding yoke. The flexible reflectors have relatively small dynamic inertias compared to the total spacecraft mass inertias; the structural mode control interaction is negligible. Therefore, the reflectors are considered to be rigid and included as a part of rigid central body in stability analysis. For the pitch dynamics of wings in the "constraint" state (see Design Considerations below for further discussion), the solar wing angular acceleration terms in Equations (4.1), (4.2), (4.4) and (4.5) may be eliminated from these equations, and the resulting model represents the linearized spacecraft dynamics with non-articulated wings.

Control Design Model of Rate Loops

The control system has an outer loop (i.e. position loop) and an inner loop (i.e. rate loop). The position loop is designed at very low bandwidth with its gain crossover well below the structural modes and consequently has generous gain margin (> 40 dB) on all flexible modes. As for the modal stability, the rate loops are the primary concerns. The design model of rate control loops in Stationkeeping Mode is given below.

Rate Gyro Dynamics

$$\theta_i^g = \frac{1}{s}\frac{a_g}{s + a_g}\omega_i \qquad (i=1,2,3)$$

Gyro Processors

$$\omega_i^g(n+1) = [\theta_i^g(n+1) - \theta_i^g(n)] / \Delta T$$

Attitude Estimators

$$\hat{\theta}_i^g(n+1) = \hat{\theta}_i^g(n) + \Delta T\, \omega_i^g(n+1)$$

$$\bar{\omega}_i^g(n+1) = \hat{\omega}_i^g(n) + \Delta\omega_i^t(n) + \Delta T\, \hat{d}_i(n) + \Delta\omega_i^c(n)$$

$$\hat{\omega}_i^g(n+1) = \bar{\omega}_i^g(n+1) + K_r [\omega_i^g(n+1) - \bar{\omega}_i^g(n+1)]$$

$$\hat{d}_i(n+1) = \hat{d}_i(n) + K_d [\omega_i^g(n+1) - \bar{\omega}_i^g(n+1)]$$

$$\Delta\omega_i^c(n) = -(H_T \Delta T / I_1)\, \hat{\omega}_3^g(n) \qquad (i=1)$$

$$= 0 \qquad (i=2)$$

$$= (H_T \Delta T / I_3)\, \hat{\omega}_1^g(n) \qquad (i=3)$$

$$\Delta\omega_i^t(n) = \Delta T\, \alpha_i^{pf}(n-1)$$

Proportional Controllers

$$\alpha_i^{pf}(z) = -N_p(z) \left(C_p \hat{\theta}_i^g + C_r \hat{\omega}_i^g\right) - \hat{d}_i(z)$$

$$N_p(z) = \frac{n_2 z^2 + n_1 z + n_0}{z^2 + d_1 z + d_0} \qquad \text{(Phase Lead Notch Filter)}$$

Control Transport Delay

$$\alpha_i^T = \frac{\Delta T}{\tau_c}\alpha_i^{pf}(n-\mu) \quad \text{if } n\Delta T + \tau_d \le t \le n\Delta T + \tau_d + \tau_c,\ \mu = (\tau_d + .5\,\tau_c)/\Delta T$$

$$= 0 \qquad \text{otherwise}$$

where

θ_i = spacecraft angular position about body axis i [deg]

ω_i = spacecraft angular rate about body axis i [deg/sec]

$\hat{\theta}_i^g$ = estimate of θ_i [deg]

ω_i^g = rate gyro measurement with calibration [deg/sec]
$\hat{\omega}_i^g$ = estimate of ω_i [deg/sec]
$\hat{d}_i$ = acceleration bias estimate about body axis i [deg/sec^2]
θ_i^g = position signal output from the rate integration gyro [deg]
RTI = real time interrupt = 16.384 msec
α_i^{pf} = filtered control acceleration commands
α_i^T = actual control acceleration acting about body axis i over τ_c [deg/sec^2]
$\Delta\omega_r^t$ = feedforward rate changes [deg/sec]
ΔT = control sample period [sec]
τ_d = control pulsing delay [sec]
τ_c = actual control pulsewidth [sec]
n = control sampling time in Stationkeeping Mode
K_r = rate estimation gain [(deg/sec)/(deg/sec)]
K_d = acceleration bias estimation gain [(deg/sec^2)/(deg/sec)]
C_p = spacecraft position control gain in Stationkeeping Mode [sec^{-2}]
C_r = spacecraft rate control gain in Stationkeeping Mode [sec^{-1}]
$N_p(z)$ = phase-lead notch filter, $z=e^{s\Delta T}$
a_g = gyro servo bandwidth [rad/sec]
I_i = spacecraft moment of inertias about body axis i [slug-ft^2]
H_T = total spacecraft angular momentum about pitch axis

It employs a discrete predictor-corrector algorithm to estimate the spacecraft rate and acceleration bias and the integration of the rate measurement (after correction for gyro bias) for position estimate. A discrete phase-lead notch filter is added in series with the attitude control acceleration command (i.e. the proportional controller excluding the acceleration bias control term, which is added to the filtered attitude control acceleration command) to provide additional phase lead for modal stabilization. The spacecraft dynamics used in the design are represented by the linearized hybrid dynamic model. The rate gyro model, which has a first order servo of 8 Hz bandwidth, output positional signal, and the gyro processor determines the rate based upon the position change over one control sample period. The equivalent transfer function of the design model will not be given in this paper. We will discuss various design concerns which are related to transient performance and structural mode stability in general, and the design philosophy in achieving the goals.

Sensitivity To Modal Parameters

The stability of the structural modes selected for baseline design may be affected by both the structural frequency uncertainty and the structural damping. A structural damping ratio of 0.0025-0.005 is added to the hybrid dynamic model. The P and Q matrices defined earlier are essentially the diagonal matrix elements of the translational and rotational rigid-flex coupling matrix B given in Equation (6.9). The coupling matrix selected for the baseline design is derived based upon the the spacecraft on orbit nominal configuration with fully deployed wings whose z-axis is directed to the Earth. By knowing the location of the interface point relative to the spacecraft CG and the wing orientation, it can be proved

that conversion of the coupling parameters to about the spacecraft CG is accomplished through the relations:

$$P_n = C_n^T P_n^o, \qquad Q_n = C_n^T Q_n^o + \tilde{R}_n P_n \tag{4.7}$$

$$P_s = C_s^T P_s^o, \qquad Q_s = C_s^T Q_s^o + \tilde{R}_s P_s \tag{4.8}$$

where "~" denotes the skew symmetric matrix operator which achieves a vector cross product. P and Q will change when the spacecraft is no longer in nominal configuration under the following two conditions. First, P and Q vary as the wings rotate about their hinge axes; and second, due to the nonlinear characteristics of panel hinge stiffness. The solar wings could be at any orientation with 16 deg or less wing separation angle. Rotation of wings will primarily affect P and Q about the spacecraft body roll and pitch axes, and almost no change about the body pitch axis in the presence of the symmetric north/south wings. Furthermore, when the east or west thrusters fire during an east/west maneuver with non-zero wing angles, the panel hinge loads as induced primarily by the linear acceleration of the spacecraft along the the panel z-axis as well as the flexibility of wings may exceed the spring preload such that the panel stiffness will drop from its hardstop region, where the nominal P and Q are derived, to the deadband region, where a soft panel stiffness is present. The worst case panel hinge loads result when the wing is at 90 deg orientation during an east/west maneuver. When this occurs, P^o and Q^o about the interface point of the wing will vary about all three axes. Both the roll and yaw control axes must be designed to stabilize all possible structural mode frequencies which may result from the rotation of wings and the nonlinear characteristics of panel stiffness.

Spacecraft CG Uncertainty

The CG offset of the spacecraft from the pressure center of maneuver thrusters will result in a thrusting disturbance about the control axis. If the actual CG offset was predicted to a 100% accuracy by the OTS in advance, the thrusting disturbance would be self-compensated with the selected thrusters -- the unique feature of the OTS. The CG uncertainty of the spacecraft has a major impact on the maneuver transients. The concerns are in two areas: (i) the CG uncertainty of the spacecraft while on station at a steady state condition, which is primarily caused by the tank misalignment and the possible imbalance of the dry spacecraft and (ii) the CG uncertainty due to propellant motion during a maneuver. The estimation errors on the acceleration bias may result in an initial transient about each axis that exceeds steady state pointing. The acceleration bias estimation gain (K_d) must be designed to minimize the maneuver initialization transient, and to avoid excitation of structural modes, while still being able to track disturbance. The transient errors can also be improved with an initialization of the acceleration bias estimates to the steady state values recorded from the last maneuver.

Firing Thrusters on Flexure

The wing flexibility may be fully excited in the steady state condition from a long maneuver burn. Also, when switching the control logic to further null the rigid body residual rates at the completion of a maneuver, an instant loss of thrusting forces could yield a significant response to the already excited flexible dynamics with the magnitude exceeding the impulse control deadband limits. To avoid firing on flexure, the rate

estimation gain (K_r) in Stationkeeping Mode must be as low as possible to filter the sensed spacecraft flexible dynamics, so that the proportional controllers determine the required control momentum based upon the estimated rigid body dynamics. The rate gain in the range $0 < K_r < 0.5$ (deg/sec)/(deg/sec) meet the requirement, while still being able to track the spacecraft dynamics to a degree of accuracy by feeding forward the commanded rate changes to the estimators from the OTS.

Control Transport Delay

Due to onboard control software processor speed, a computational delay on rate change command processing in OTS is induced. Such a delay together with the phase delay induced by the gyro rate signal processing and the control pulsewidth delivery induce a phase lag to each control loop, which affects the stability of structural modes. To compensate for the loss of phase due to the control transport delay, a phase-lead notch filter is employed in Stationkeeping Mode to provide each control loop with an additional phase lead. The design philosophy is to set the modulation frequency so that the half sample frequency is well above the dominant modes to ensure phase stabilization of these modes with additional phase lead produced by the notch filter. Because the notch filter is not wide enough to provide all modes with sufficient phase lead, the modes near the half sample rate are gain-stabilized. Phase stabilization of the dominant modes simply means that the control loop will generate a stabilizing feedback signal to that mode. This is to be contrasted to gain stabilization wherein the non-dominant modes rely upon the structural damping of the spacecraft to provide enough damping to overcome any slight destabilizing effects.

Effect of SWD Deadband on Torsional Mode Stability

Due to the solar wing drive (SWD) backlash, the flexible pitch dynamics of wings about their interface points may appear in one of the three states: "free-free", "constraint" and one between these two states, depending upon whether the wings are inside or outside the deadband and the magnitude of friction. The SWD has a 0.5^{o} deadband. When the wings are inside the deadband and the magnitude of friction is insufficient to overcome the wing relative motion, the central body pitch dynamics are then disturbed by the load torques with a phase-shifted bang-bang profile whose magnitude is equal to Coulomb friction. It is very complex to analyze the torsional mode stability with such a profile. The best way to examine the performance is through the simulation by actually including the SWD. When the wings are inside the deadband but the friction is negligible so that the flexible dynamics of each wing are "free-free" about its pitch axis, the central body pitch dynamics are then fully decoupled from the wing relative motion. The central body under this state is considered to be rigid. The stabilities of the "free-free" modes will not become a problem as long as the free-free motion remains inside the deadband. Still, when inside the deadband but the relative motion of the wing is locked up by friction, or, the wing-to-body rates are large enough to break the friction and the gear teeth are recontacted to the SWD shaft, the pitch flexible dynamics of wing are now considered to be "constrained" about its pitch axis; that is, the SWD will output load torques in absence of stepping commands, to drive the wings to prevent them from moving about the pitch axis of the central body. From this viewpoint, the wings are also fixed to the central body about its pitch axis and the flexible dynamics of wings are fully coupled into the body pitch axis through the load torques.

Plume Impingement Effects

Forces and torques created by thruster plume impingement on spacecraft appendages (north wing and east/west antennas) may result in a net change in the overall control torques and an unmodelled excitation of the structural modes. From the spacecraft stability viewpoint, the principle concern is that the plume disturbance shall not add phases to erode the phase margins of the phase stabilization modes nor magnify amplitudes to degrade the gain margins of the gain stabilization modes.

Since $\bar{r}$ is the pressure center of plume forces, the plume induced torques about the deformed S/C CG is

$$\bar{T}^p = (\bar{R}_n + \bar{r}) \times \bar{F}^p + \Delta\bar{T}^p$$

$$\Delta\bar{T}^p = \bar{T}^p - (\bar{R}_n + \bar{r}) \times \bar{F}^p = \{s\}^T \left\{ C_n T^p - (\widetilde{C_n R_n} + \tilde{r})(C_n F^p) \right\}$$

or

$$\Delta T^p = C_n T^p - (\widetilde{C_n R_n} + \tilde{r})(C_n F^p)$$

Referring to Equation (4.5), the modal excitation of the north wing is induced by (i) the spacecraft and the north wing motion driven by the terms

$$\varepsilon_n = P_n^T \ddot{x} + Q_n^T \dot{\omega} + (e_2^T Q_n^p)^T \dot{\omega}_n$$

and (ii) the plume disturbance

$$\varepsilon^p = \Phi^T \left[(C_n F^p)^T (\Delta T^p)^T \right]^T$$

acting about point i, the assumed pressure center of plume forces.

In the presence of the plume disturbance ε^p acting on the North wing, the open loop dynamics transfer function $\omega(s)/T^t(s)$ from an impulse response can be derived from the single axis hybrid dynamic model with one mode only as

$$\frac{\omega(s)}{T^t(s)} = \frac{1}{I_{s/c}} \frac{1-\mu_1}{1-\mu_2} \frac{s^2 + 2\zeta_1\lambda_1 s + \lambda_1^2}{s^2 + 2\zeta_2\lambda_2 s + \lambda_2^2} \frac{1}{s}$$

where

$$\mu_1 = \varepsilon^p q_n / T^t$$
$$\mu_2 = 2 q_n^2 / I_{s/c}$$
$$\zeta_1 = \zeta / \sqrt{1-\mu_1}$$
$$\zeta_2 = \zeta / \sqrt{1-\mu_2}$$
$$\lambda_1 = \lambda / \sqrt{1-\mu_1}$$
$$\lambda_2 = \lambda / \sqrt{1-\mu_2}$$

The parameters ε^P and q_n are the plume disturbance and the rigid-flex rotational coupling term associated with the structural mode of frequency λ and modal damping ζ. T^t is the impulse thrusting torque about the single axis in concern. If $\mu_1 \neq 0$ (i.e. $\varepsilon^P \neq 0$), the plume disturbance will perturb the zeros of the transfer function above from their nominal locations. If the perturbed zeros move toward the poles (i.e. $\mu_1 > 0$) , then it improves the margin of the phase-stabilized mode, producing less rigid-flex coupling. On the contrary, if $\mu_1 < 0$, the zeros move away from the poles, and the loop gain is magnified by a factor of $(1- \mu_1)$: both of these factors will erode the margin of any gain-stabilized mode.

The geometry of the thrusters and solar wings are such that, in fundamental modes (those with no inflection points), the plume impingement coupling and the rigid-flex dynamic coupling act in phase with one another to excite a mode. In other words, plume impingement acts to amplify modal excitation already present due to rigid-flex dynamic coupling. This implies that $\mu_1 < 0$ for fundamental modes. In this case, the zeros of the transfer function move even further from the poles, exacerbating the flexible dynamics coupling problem for these modes. Thus if $\mu_1 < 0$, it is desirable to have the magnitude of μ_1 as small as possible: $|\mu_1| \ll |\mu_2|$ is goodness. In this case study, $\mu_1 = -0.0388$ for the first out-of-plane mode at 0.1185 Hz. Fortunately, values are small in comparison to the corresponding $\mu_2 = 0.7037$, indicating that plume impingement is not a dominant effect. One measure is the zero/pole frequency ratio: with no plume impingement

$$\frac{\lambda_1}{\lambda_2} = \sqrt{1-\mu_2} = \sqrt{1-.7037} = .544$$

whereas with plume impingement

$$\frac{\lambda_1}{\lambda_2} = \sqrt{\frac{1-\mu_2}{1-\mu_1}} = \sqrt{\frac{1-.7037}{1+.0388}} = .534$$

This is a change of only 1.87% which is small relative to the 5% or 10% accuracy to which μ_2 is known to begin with (A 1.58% change in the value of μ_2 would result in the same change in zero/pole ratio).

Higher frequency modes whose mode shapes include an odd number of inflection points between the attach point and the "point of application" of the plume impingement force can exhibit a positive value for μ_1. In this case, the plume impingement force acts opposite to the direct rigid-flex dynamic coupling and tends to reduce modal excitation. In the transfer function this is reflected by the fact that the zeros move closer to the poles, thus tending to cancel. Should μ_1 ever get as big as μ_2, the zero would exactly cancel. An even higher value of μ_1 would reverse the phase of the modal coupling. This situation is of less interest, and is probably not possible with plume impingement as the excitation source. Since plume impingement is, in reality, a distributed force rather than a point force as modeled herein, its viability in exciting a higher frequency mode diminishes rapidly as the

number of inflection points increases. The assumption made herein that plume impingement force is applied at a single point loses its validity for higher frequency modes, therefore results should not be taken too literally for such modes. The "constrained" state is similar to those for roll or yaw loop.

5. Control Loop Stability

With the panel support cantilevered at its base about the transverse axes, but free in torsion, Table 5.1 characterizes the flexibility of 0 deg, 3-panel single wing in terms of the modal frequencies with associated dynamic inertias about the interface point of wing. The first twelve modes as listed contain > 99% of the total inertias of wing about each axis, which are sufficient to describe the flexible characteristics of wings. When the wings are attached to their base, the flexible dynamics appeared to the angular motion of the spacecraft through rigid-flex coupling have frequencies higher than that of the cantilever modes. The increased modal frequencies, assuming perfectly symmetric wings, are defined as the system modes in Table 5.1. The frequencies of the system modes will shift as the wings rotate about the hinge axes; 90 deg wings yield out-of-plane cantilever modes in the yaw axis and in-plane cantilever modes in the roll axis. For the transfer function of the corresponding open loop dynamics, the dominant modes have wider pole/zero separations. One of the design goals is to stabilize the system modes under any wing orientation.

Table 5.1. Solar Wing Structural Modes at Zero Degree Wing Angle

Mode	Frequency, Hz			Dynamic Inertia (about interface point), $Kg-m^2$	Definition
	Cantilever	System			
		Constraint	Free-Free		
1	0.1185	0.2117		44.498	out-of-plane
2	0.3547	0.6323		45.970	in-plane
3	0.7051	0.7631		8.722	out-of-plane
4	0.8508	0.8582	1.3645	4.498	torsional
5	1.8254	1.8627		3.550	out-of-plane
6	2.8058	2.8091	3.1938	1.588	torsional
7	3.1668	3.2573		3.412	out-of-plane
8	5.1320	5.1338	5.3698	0.854	torsional
9	6.1608	6.1704		0.592	out-of-plane
10	8.2235	8.2333		0.432	out-of-plane
11	8.9668	8.9898	9.1631	0.643	torsional
12	9.4732	9.4882	9.7306	0.524	torsional

Figure 5.1 shows the discrete-time Bode plots and Nichols chart of the 8*RTI Stationkeeping Mode spacecraft roll rate control loop with no structural filter or control transport delay. The control bandwidth was designed to limit transient errors to within 0.1 deg in the presence of a 1.5 inches spacecraft CG offset along the z-axis. The first five out-of-plane modes at nominal frequencies, 0.5% structural damping and 0 deg wing angle were included. Using gyro references, both the first (mode 1) and second out-of-plane mode (mode 3) are phase-stabilized with about 70 deg and 30 deg phase margins, respectively, and the remaining out-of-plane modes (5, 7, 9 & 10) are gain-stabilized with at least 22 dB gain margin. The control design provides a 6.5:1 ratio to the separation between the zero gain crossing frequency and the pole of the first structural mode. Figure 5.2 shows the same design without a structural filter, but with a 2 RTI control transport delay. The transport delay effect can be seen on the structural modes greater than 0.33 Hz, to which the phase lag induced by transport delay was added, yielding almost no phase margin on the second out-of-plane mode. The linear design was then improved with a phase-lead notch filter, which has a unit gain in the low frequency range and a maximum of 68 deg phase lead at the notch frequency of 1.209 Hz. Figure 5.3 shows that the phase margin of the second out-of-plane mode was increased up to 36 deg with the phase-lead notch filter. The notch frequency was carefully selected to ensure that all phase-stabilized modes will remain in the phase stabilization region in the presence of 100% frequency increase as shown in Figure 5.4. Although a 100% frequency increase is allowed before mode 3 loses its phase stabilization characteristics, this mode is also gain stabilized once its frequency increases from the nominal. Also, the third out-of-plane mode (mode 5) is gain stabilized with 13 dB margin at its nominal frequency (1.8627 Hz, system mode). Decreasing the frequency of this mode immediately leads it to the phase stabilization region, while the gain stabilization characteristics are still retained. This mode reaches adequate phase margin before the gain stabilization characteristics vanish at more than 50% frequency drop as shown in Figure 5.5.

The pitch transient during south maneuvers is affected by thrusting disturbance, primarily induced by both the thruster cant angles and the thrust mismatch. The net pitch disturbance is estimated to be 0.4 ft-lb, which requires the 8 msec minimum control pulsewidth to be fired at a rate of 1.667 Hz at which the half sample rate is nearly equal to the first torsional mode. The pitch loop has rigid response in south stationkeeping because the effect of the torsional mode is insignificant about the half sample control rate. The pitch transient during east/west maneuvers is primarily affected by the S/C CG yaw offset from the pressure center of the maneuver thrusters. With a 5.9 inches yaw offset, it requires a 1.5 RTI control pulsewidth to be fired every modulation period. The linear frequency analysis of the spacecraft pitch rate loop in Stationkeeping Mode when the pitch wing dynamics with SWD are in the "constraint" state is shown in Figure 5.6. The design with the same phase-lead notch filter as applied to the roll and yaw loops was based on 2 RTIs control pulsewidth, 2 RTIs control transport delay and 8 RTIs modulation period. The first torsional mode (mode 4) is both phase and gain stabilized at its nominal frequency with $55 sup o$ phase margin and 20 dB gain margin. Either dropping or increasing the frequency produces no impact on stability at all.

Figure 5.7 shows the stability of the first in-plane mode (mode 2) at nominal frequency and with the phase-lead notch filter. It is also phase-stabilized with 46 deg phase margin. This mode remains in the phase stabilization region even with a 100% frequency increase while still having 23^{o} phase margin as shown in Figure 5.8.

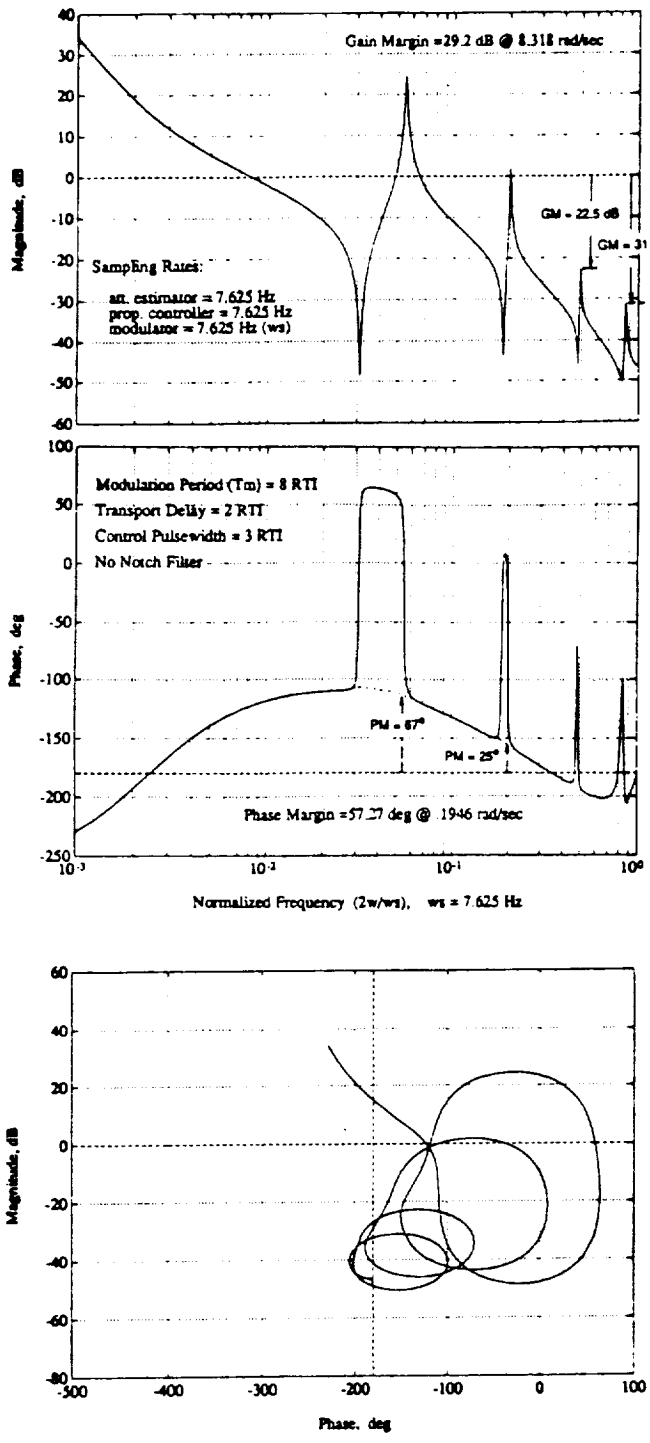


Figure 5.1 Discrete Bode Plots and Nichols Chart of Roll Rate Loop in Stationkeeping Mode at 0.5% Structural Damping, Nominal Frequencies : without Structural Filter and Zero Control Transport Delay

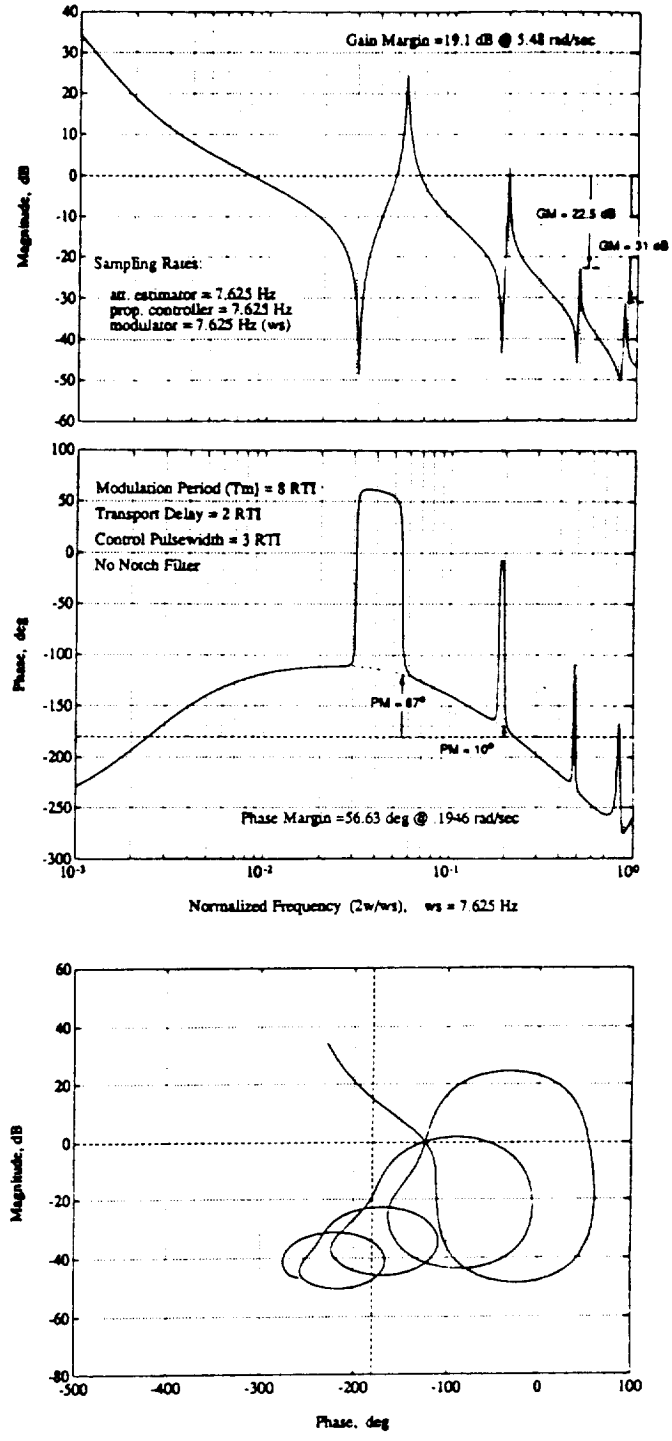


Figure 5.2 Discrete Bode Plots and Nichols Chart of Roll Rate Loop in Stationkeeping Mode at 0.5% Structural Damping, Nominal Frequencies : without Structural Filter, but with 2 RTIs Control Transport Delay

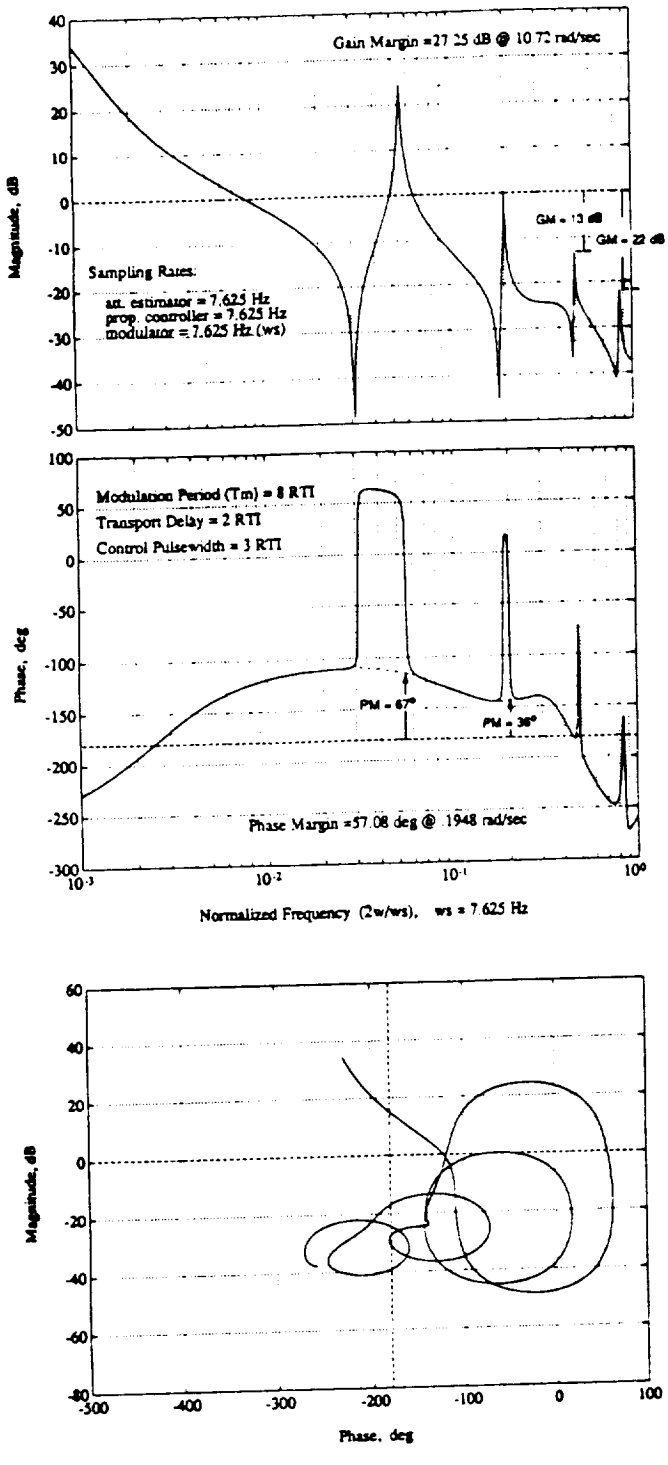


Figure 5.3 Discrete Bode Plots and Nichols Chart of Roll Rate Loop in Stationkeeping Mode at 0.5% Structural Damping, Nominal Frequencies : with Phase-Lead Notch Filter and 2 RTIs Control Transport Delay

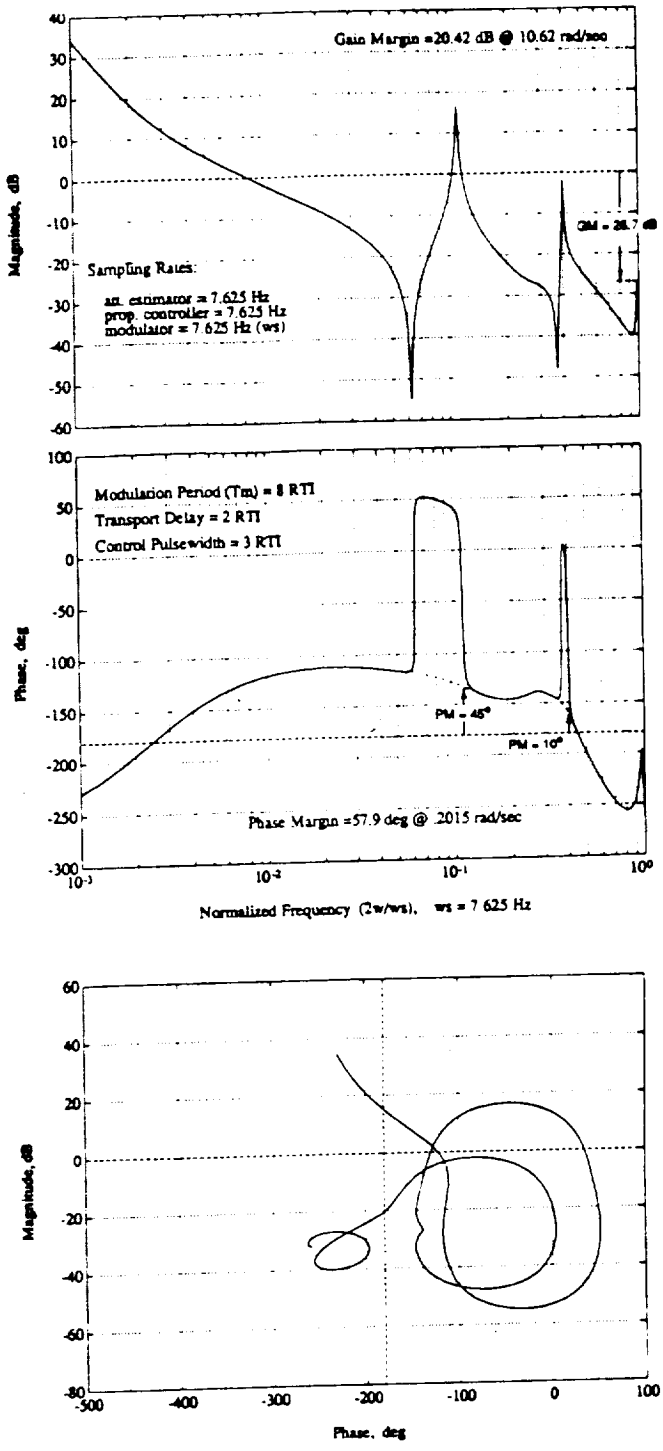


Figure 5.4 Discrete Bode Plots and Nichols Chart of Roll Rate Loop in Stationkeeping Mode at 0.5% Structural Damping, 100% Frequency Increase : with Phase-Lead Notch Filter and 2 RTIs Control Transport Delay

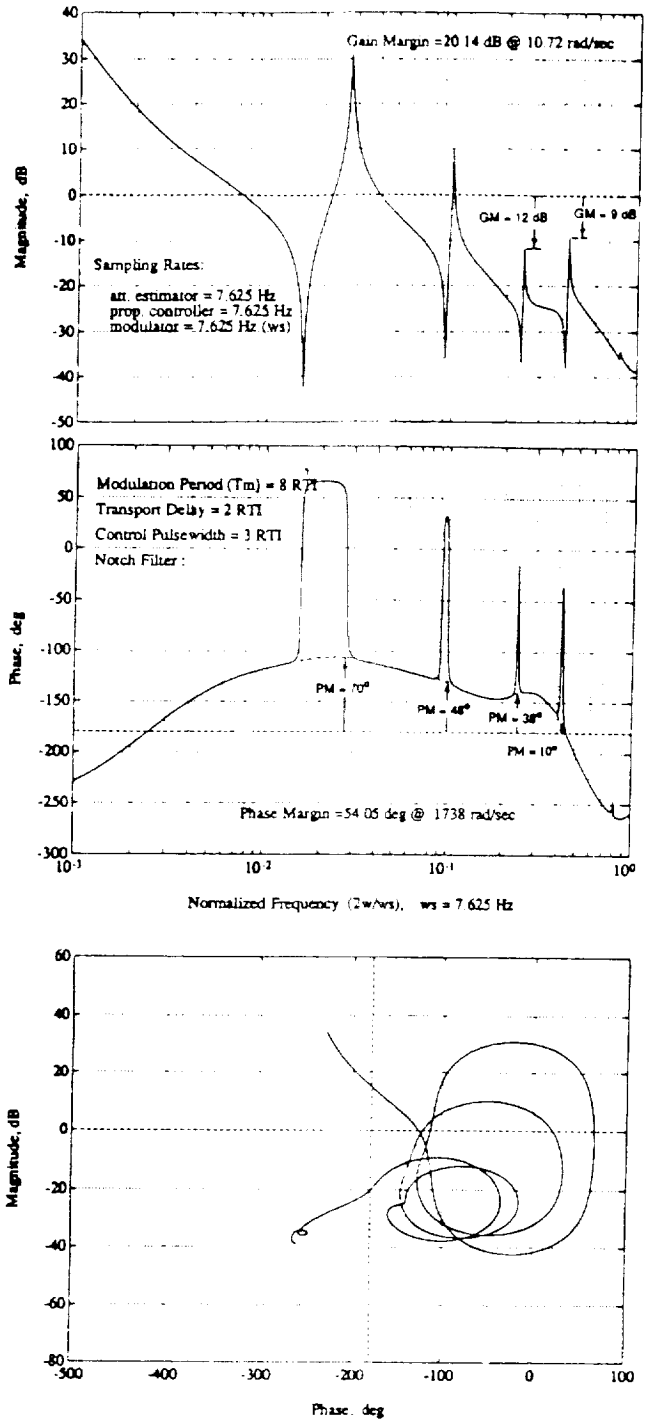


Figure 5.5 Discrete Bode Plots and Nichols Chart of Roll Rate Loop in Stationkeeping Mode at 0.5% Structural Damping, 50% Frequency drop : with Phase-Lead Notch Filter and 2 RTIs Control Transport Delay

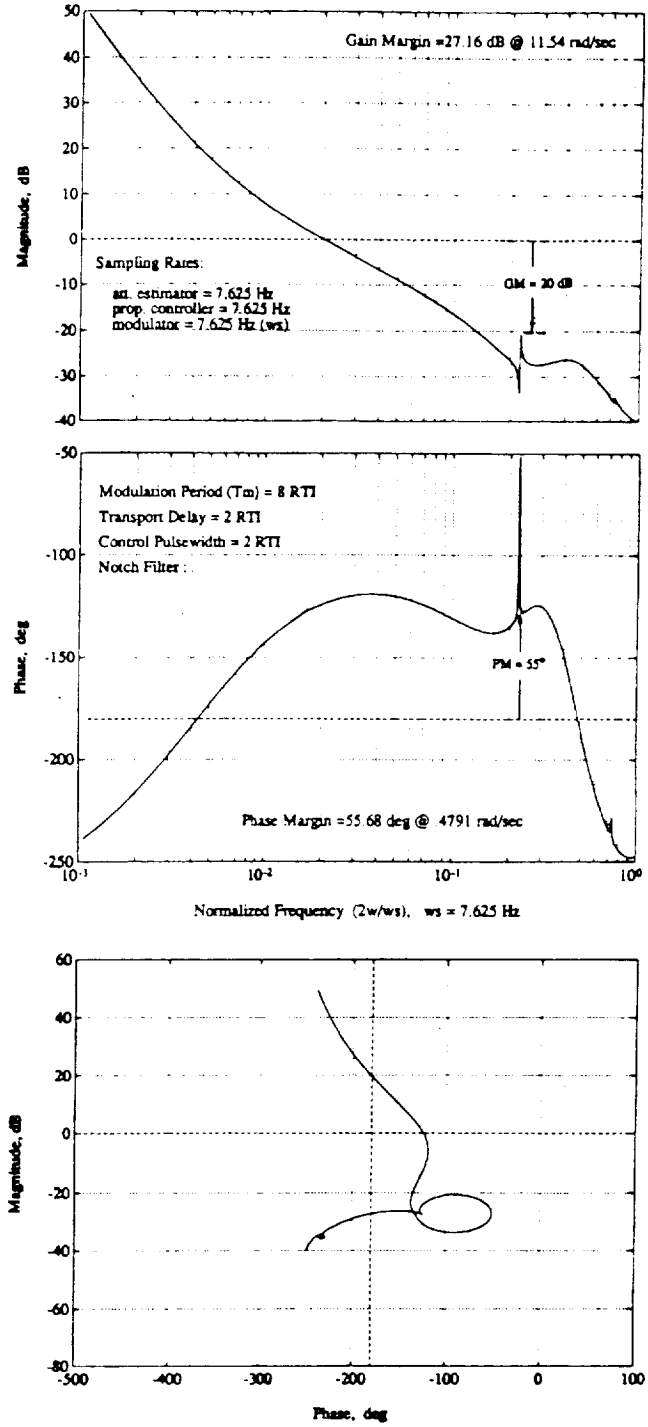


Figure 5.6 Discrete Bode Plots and Nichols Chart of Pitch Rate Loop in Stationkeeping Mode at 0.5% Structural Damping, Nominal Frequencies : with Phase-Lead Notch Filter and 2 RTIs Control Transport Delay

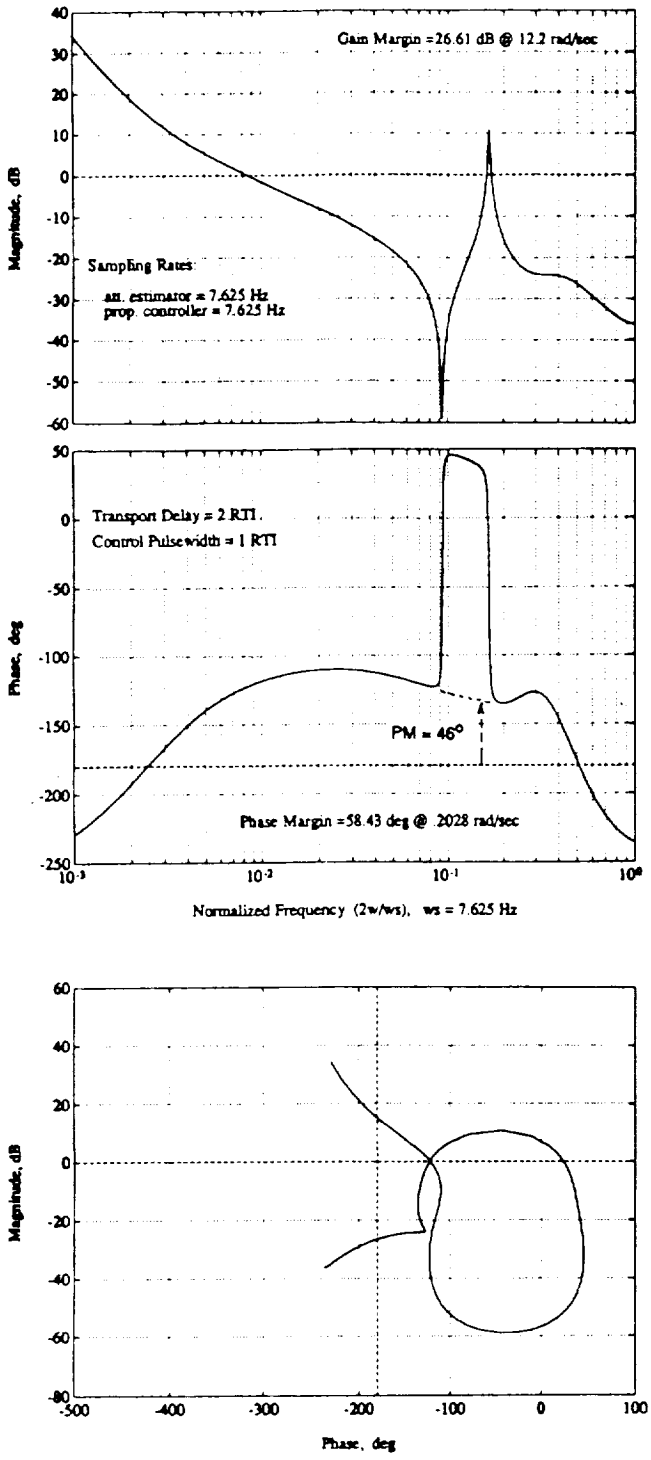


Figure 5.7 Discrete Bode Plots and Nichols Chart of Yaw Rate Loop in Stationkeeping Mode at 0.5% Structural Damping, Nominal Frequencies : with Phase-Lead Notch Filter and 2 RTIs Control Transport Delay

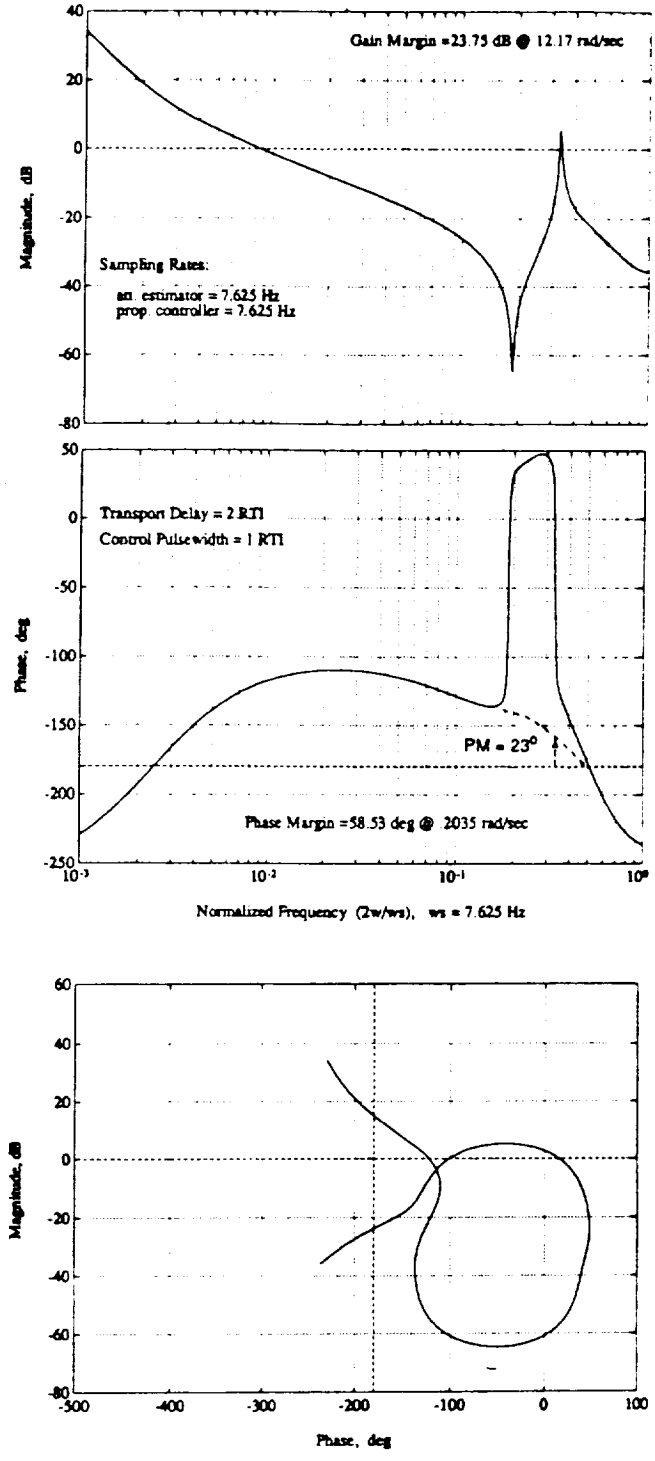


Figure 5.8 Discrete Bode Plots and Nichols Chart of Yaw Rate Loop in Stationkeeping Mode at 0.5% Structural Damping, 100% Frequency Increase : with Phase-Lead Notch Filter and 2 RTIs Control Transport Delay

6. Dynamic Model Validation and Digital Implementation

The main objective is to consider issues dealing with flexibility in multibody dynamics. Multibody dynamics is differentiated from structural dynamics by its capability to undergo arbitrary rigid body motion. Analysis of flexible structures are well established using finite element method within the context of structural dynamics. In multibody dynamics context, need for flexibility modelling arose in recent years as exemplified by large space structures and 3-axes stabilized satellites.

To systematically address the addition of flexible domain, the virtual work principle is chosen as the basis for derivation. The motivation for this choice is based on intended discretization using the finite element method. By choosing the same basis for multibody dynamics and for the finite element method, extensions into nonlinear flexibility is natural and consistent. Other choices are readily available in the literature [1-3].

Virtual Work Principle

An integral representation of the governing equations of motion of solids are imbedded in the virtual work principle. By deriving the multibody dynamics equations via the virtual work principle, a consistent treatment of flexible domain can be made. In practice, the flexible domain is discretized using the finite element method. The virtual work principle is the basis for the finite element method. The technology developed in the finite element method can be integrated into the flexible multibody dynamics efforts.

The virtual work principle states

$$\delta W_{ext} = \int_v \delta \mathbf{R} \cdot \left(\mathbf{f} - \rho \ddot{\mathbf{R}}\right) dV = \delta W_{int} = \int_v \delta \varepsilon : \sigma \, dV \tag{6.1}$$

where

$\mathbf{R}$ = material particle position vector wrt inertial frame
$\mathbf{f}$ = force/unit volume
ρ = density
$\mathbf{V}$ = reference configuration
ε = strain
σ = stress

The main advantages offered by applying the virtual work principle are twofold. First, the integral representation together with the virtual displacements allow domain decomposition between the rigid and the flexible portions of a vehicle. Second, a consistent formulation of a flexible multibody vehicle can be derived and assessed. Consistency refers to final discretization using the finite element method. Once such consistent derivation is made, extensions to nonlinear flexible models can be made by adopting techniques developed in the finite element method [14].

Rigid Body with Attached Flexible Appendage

To further explore the method presented by the virtual work principle, an idealized flexible spacecraft model is derived. The idealization involve representing the spacecraft as a rigid body with attached flexible appendage. The flexible appendage is assumed to be fixed to the rigid body. Articulation is not allowed. Even with this simplifying assumptions, a wide class of vehicles can be modelled.

Consider the idealization shown in Figure 6.1.

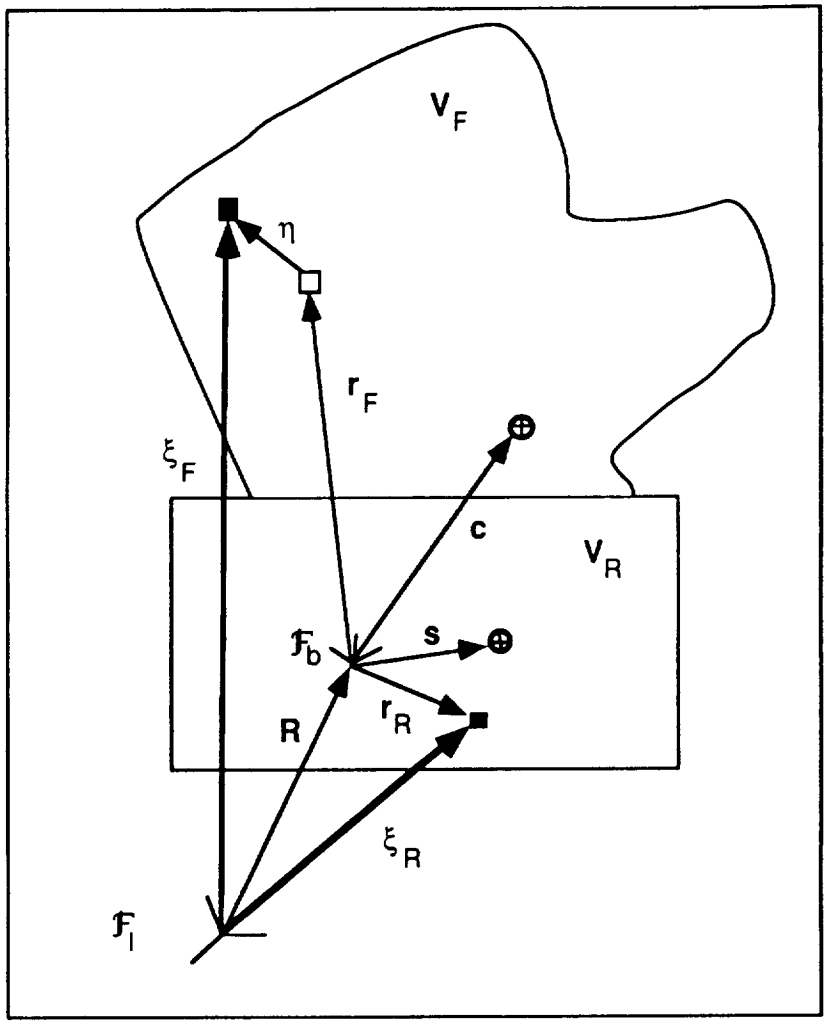


Figure 6.1. Idealized Rigid Body with Flexible Appendage.

The domains, frames, and the position vectors are defined as

V_F = flexible domain
V_R = rigid domain
$\mathcal{F}_I$ = inertial frame

$\mathcal{F}_b$ = body frame
$\mathbf{R}$ = inertial frame origin to body frame origin
ξ_R = inertial frame origin to rigid body material particle
$\mathbf{r}_R$ = body frame origin to rigid body material particle
$\mathbf{s}$ = body frame origin to rigid body center of mass
ξ_F = inertial frame origin to flexible material particle
$\mathbf{r}_F$ = body frame origin to reference material particle position
η = relative particle displacement
$\mathbf{c}$ = body frame origin to vehicle center of mass

Application of the virtual work principle to this vehicle yields

$$\int_{V_R} \delta\xi_R \cdot \left(\mathbf{f} - \rho\ddot{\xi}_R\right) dV + \int_{V_F} \delta\xi_F \cdot \left(\mathbf{f} - \rho\ddot{\xi}_F\right) dV = \int_{V_F} \delta\epsilon : \sigma \, dV \tag{6.2}$$

with

$$\xi_R = \mathbf{R} + \mathbf{r}_R$$

$$\ddot{\xi}_R = \mathcal{F}_b^T \left[\dot{\underline{u}} + \underline{\omega} \times \underline{u} + \dot{\underline{\omega}} \times \underline{r}_R + \underline{\omega} \times \left(\underline{\omega} \times \underline{r}_R\right)\right] \equiv \mathcal{F}_b^T \underline{a}_R$$

$$\delta\xi_R = \mathcal{F}_b^T \left[\delta\underline{x} + \delta\underline{\theta} \times \underline{r}_R\right]$$

and

$$\xi_F = \mathbf{R} + \mathbf{r}_F + \eta$$

$$\ddot{\xi}_F = \mathcal{F}_b^T \left[\dot{\underline{u}} + \underline{\omega} \times \underline{u} + \dot{\underline{\omega}} \times (\underline{r}_F + \underline{\eta}) + 2\,\underline{\omega} \times \dot{\underline{\eta}} + \underline{\omega} \times \left(\underline{\omega} \times (\underline{r}_F + \underline{\eta})\right) + \ddot{\underline{\eta}}\right] \equiv \mathcal{F}_b^T \underline{a}_F$$

$$\delta\xi_F = \mathcal{F}_b^T \left[\delta\underline{x} + \delta\underline{\theta} \times (\underline{r}_F + \underline{\eta}) + \delta\underline{\eta}\right]$$

where

$\underline{u}$ = velocity of body frame wrt inertial frame
$\underline{\omega}$ = angular rate of body frame wrt inertial frame
$\delta\underline{x}$ = virtual displacement of the body frame
$\delta\underline{\theta}$ = virtual rotation of the body frame
$\delta\underline{\eta}$ = virtual relative displacement

The components, underlined, are defined with respect to the first occurrence of the frame definition. For example, in above definition, the components $\underline{u}$ and $\underline{\omega}$ are defined in the body frame. For further discussion on this notation is clearly covered in [1]. Note that the virtual quantities are obtained through infinitesimal variation of the current equilibriated state. Substituting the above quantities into Equation (6.1), following three sets of equations, with respect to the body frame, can be derived.

$$\underline{F} = m\dot{\underline{u}} + m\,\underline{\omega}\times\underline{u} + m\,\dot{\underline{\omega}}\times\underline{c} + m\,\underline{\omega}\times(\underline{\omega}\times\underline{c}) + 2\,\underline{\omega}\times\left(\int_{v_F}\dot{\underline{\eta}}\,\rho\,dV\right) + \left(\int_{v_F}\ddot{\underline{\eta}}\,\rho\,dV\right) \tag{6.3}$$

$$\underline{T} = m\,\underline{c}\times\dot{\underline{u}} + m\,\underline{c}\times(\underline{\omega}\times\underline{u}) + I\,\dot{\underline{\omega}} + \underline{\omega}\times I\,\underline{\omega} + 2\int_{v_F}(\underline{r}_F + \underline{\eta})\times(\underline{\omega}\times\dot{\underline{\eta}})\,\rho\,dV + \int_{v_F}(\underline{r}_F + \underline{\eta})\times\ddot{\underline{\eta}}\,\rho\,dV \tag{6.4}$$

$$\begin{aligned}\int_{v_F}\delta\underline{\eta}^T\underline{f}_F\,dV &= \left(\int_{v_F}\delta\underline{\eta}^T\rho\,dV\right)\dot{\underline{u}} + \left(\int_{v_F}\delta\underline{\eta}^T\rho\,dV\right)\underline{\omega}\times\underline{u} \\ &+ \left(\int_{v_F}\delta\underline{\eta}^T\,\dot{\underline{\omega}}\times(\underline{r}_F + \underline{\eta})\,\rho\,dV\right) + \left(\int_{v_F}\delta\underline{\eta}^T\,\underline{\omega}\times(\underline{\omega}\times(\underline{r}_F + \underline{\eta}))\,\rho\,dV\right) \\ &+ 2\left(\int_{v_F}\delta\underline{\eta}^T(\underline{\omega}\times\dot{\underline{\eta}})\,\rho\,dV\right) + \left(\int_{v_F}\delta\underline{\eta}^T\,\ddot{\underline{\eta}}\,\rho\,dV\right) + \left(\int_{v_F}\delta\varepsilon : \sigma\,dV\right)\end{aligned} \tag{6.5}$$

with

$$\underline{F} = \underline{F}_b + \int_{v_F}\underline{f}_F\,dV$$

$$\underline{T} = \underline{T}_b + \int_{v_F}(\underline{r}_F + \underline{\eta})\times\underline{f}_F\,dV$$

where

$\underline{F}_b$	=	force applied to the rigid body at body frame origin
$\underline{T}_b$	=	torque applied to the rigid body about body frame origin
I	=	instantaneous vehicle inertia matrix wrt body frame origin

Often in practice, the integral representation in Equations (6.3) - (6.5) is skipped by assuming the lumped mass idealization. However, the discretization of the flexible domain into finite element idealization stem from these equations. The lumped mass idealization is an extreme case. Such inconsistent assumption with finite element method may produce inaccurate results for crude finite element mesh. More systematic study should be made to assess the consequence of such assumption.

As closure, the lumped mass idealization will be made to produce a set of equations that may be compared to previous derivation [4]. The lumped mass idealization takes the volume integral and cast it into a sum spanning the total number of nodes in a finite element mesh. For an arbitrary function, this idealization can be expressed as

$$\int_{V_F} f(\underline{r}, \dot{\underline{r}}, \ddot{\underline{r}})\,\rho\, dV = \sum_i f(\underline{q}_i, \dot{\underline{q}}_i, \ddot{\underline{q}}_i)\, m_i$$

where

$\underline{q}_i$ = finite element nodal displacement vector
m_i = corresponding lumped mass

Adopting this idealization, Equations (6.3) - (6.5) can be reduced to

$$\underline{F} = m\dot{\underline{u}} + m\,\underline{\omega}\times\underline{u} + m\,\dot{\underline{\omega}}\times\underline{c} + m\,\underline{\omega}\times(\underline{\omega}\times\underline{c}) + 2\sum_i m_i\,\underline{\omega}\times\dot{\underline{q}}_i + \sum_i m_i\,\ddot{\underline{q}}_i \tag{6.6}$$

$$\underline{T} = m\,\underline{c}\times\dot{\underline{u}} + m\,\underline{c}\times(\underline{\omega}\times\underline{u}) + I\,\dot{\underline{\omega}} + \underline{\omega}\times I\,\underline{\omega} + 2\sum_i m_i\,(\underline{r}_i + \underline{q}_i)\times(\underline{\omega}\times\dot{\underline{q}}_i) + \sum_i m_i\,(\underline{r}_i + \underline{q}_i)\times\ddot{\underline{q}}_i \tag{6.7}$$

$$\underline{f}_i = m_i\dot{\underline{u}} + m_i\,\underline{\omega}\times\underline{u} + m_i\,\dot{\underline{\omega}}\times(\underline{r}_i + \underline{q}_i) + m_i\,\underline{\omega}\times(\underline{\omega}\times(\underline{r}_i + \underline{q}_i)) + 2\,m_i\,\underline{\omega}\times\dot{\underline{q}}_i + m_i\,\ddot{\underline{q}}_i + \sum_j K_{ij}\,\underline{q}_j \tag{6.8}$$

where

m = total vehicle mass
K_{ij} = assembled stiffness matrix

By interpreting the stiffness matrix as the tangent stiffness matrix, the equations are valid for nonlinear flexible systems. Since modal reduction generally is not possible for nonlinear flexible systems, the finite element nodal degrees of freedom must be used to represent flexible degrees of freedom. For linear flexible system, an indepth coverage of an alternate derivation of Equations (6.6) - (6.8) is provided in [4].

Extending Symbolic Rigid Body Code to include Flexibility

In the recent years symbolic manipulation software capable of generating rigid body code became available. Some example of such codes are SD/FAST (Symbolic Dynamics, Inc.) [6], AUTOLEV (OnLine Dynamics, Inc.) [8], and AUTOSIM (Univ. of Michigan) [9]. For rigid vehicles, these tools can dramatically reduce the time spent on deriving and implementing the equations of motion.

By combining the codes generated by the symbolic manipulation software with reduced set of "hand" derived equations addressing the flexible domain, the capability of these codes can be extended to flexible vehicles. A systematic method for such an extension is provided for a satellite class of flexible vehicles. This method will be

illustrated with previously derived equations of motion for a rigid body with flexible appendages.

For a rigid body with flexible appendages, the resulting equations can be partitioned into

$$\begin{bmatrix} A & B \\ B^T & 1 \end{bmatrix} \begin{Bmatrix} \ddot{u}_R \\ \ddot{u}_F \end{Bmatrix} = \begin{Bmatrix} R_R + R_{RF} \\ R_F \end{Bmatrix} \tag{6.9}$$

where

u_R = rigid degrees of freedom
u_F = modal amplitude degrees of freedom

and

$$B = [P \quad Q]$$

Note that the P and Q submatrices are defined in the previous section. In Equation (6.9), a modal reduction has been assumed. The portion of the partitioned equation generated by a symbolic manipulation software is

$$[A] \{\ddot{u}_R\} = \{R_R\} \tag{6.10}$$

This portion is obtained by supplying the symbolic manipulation software information on the current configuration. In another words, the total vehicle is assumed to be rigid. The requirement of current configuration entails configuration update at each integration step. The current configuration is the reference configuration.

The solution process follows by forming

$$\{\ddot{u}_F\} = \{R_F\} - [B^T] \{\ddot{u}_R\} \tag{6.11}$$

and substituting into the rigid partition to yield

$$[A - BB^T] \{\ddot{u}_R\} = \{R_R + R_{RF}\} - [B] \{R_F\} \tag{6.12}$$

In terms of actual equations of motion, by observing the structure of Equations (6.4) - (6.6), the necessary additional partitions can be generated by discretization and modal reduction of the terms

$$\begin{Bmatrix} \left(\int_{v_F} \ddot{\underline{\eta}}\,\rho\,dV\right) \\ \left(\int_{v_F} (\underline{r}_F + \underline{\eta}) \times \ddot{\underline{\eta}}\,\rho\,dV\right) \end{Bmatrix} \rightarrow [B]\,\{\ddot{u}_F\} \tag{6.13}$$

$$\begin{Bmatrix} \left(\int_{v_F} \underline{f}_F\,dV\right) \\ \left(\int_{v_F} (\underline{r}_F + \underline{\eta}) \times \underline{f}_F\,dV\right) \end{Bmatrix} - \begin{Bmatrix} 2\,\underline{\omega} \times \left(\int_{v_F} \dot{\underline{\eta}}\,\rho\,dV\right) \\ \left(2\int_{v_F} (\underline{r}_F + \underline{\eta}) \times (\underline{\omega} \times \dot{\underline{\eta}})\,\rho\,dV\right) \end{Bmatrix} \rightarrow \{R_{RF}\} \tag{6.14}$$

The flexible partition can be generated by Equation (6.5).

Methodology presented above produces an "exact" set of equations. Standard assumptions such as constant vehicle center of mass and inertia together with small relative flexible displacements can be made as deemed plausible to reduce computational effort.

Rigid Body with Articulated Flexible Appendages

With assumption that an symbolic manipulation software will be used to generate the rigid partition of the equation of motion, only the required matrices for the articulated flexible domain will be documented. The idealized articulated flexible appendage is shown in Figure 6.2.

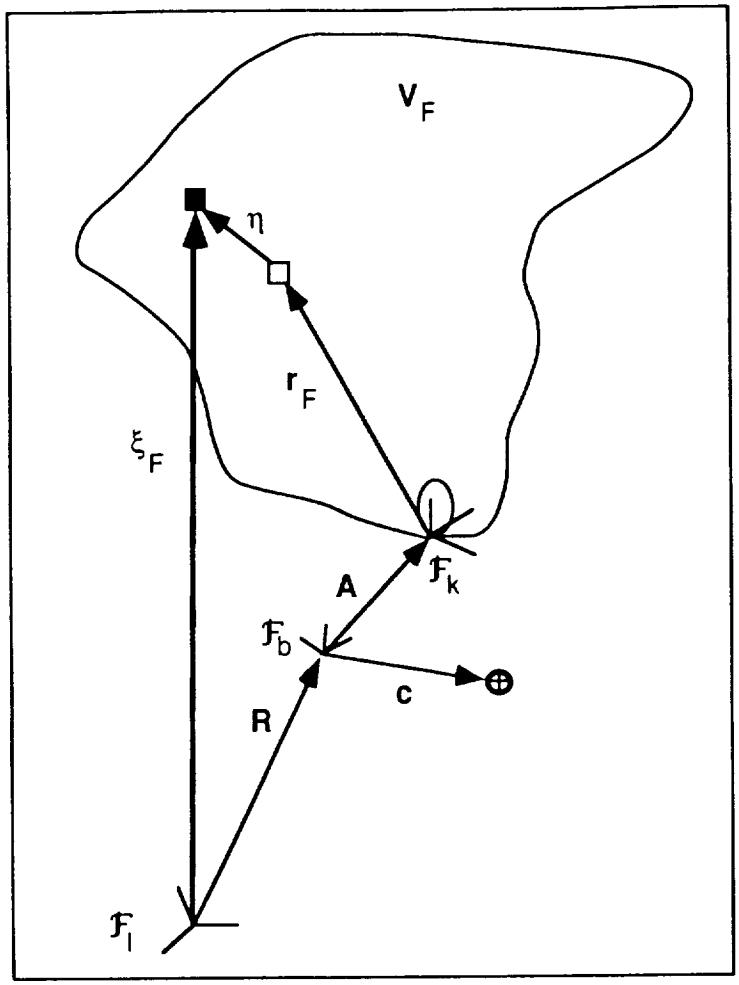


Figure 6.2. Idealized Articulated Flexible Appendage.

The quantities are defined as

V_F	=	flexible domain
$\mathcal{F}_I$	=	inertial frame
$\mathcal{F}_b$	=	body frame
$\mathcal{F}_k$	=	appendage frame imbedded in the yoke body
R	=	inertial frame origin to body frame origin
A	=	body frame origin to appendage frame origin
ξ_F	=	inertial frame origin to flexible material particle
$\mathbf{r}_F$	=	appendage frame origin to reference material particle position
η	=	relative particle displacement
c	=	body frame origin to vehicle center of mass

In generating the rigid partition using a symbolic manipulation software, consider the yoke body and the flexible domain as a single rigid body defined in the current configuration.

Application of the virtual work principle to the flexible appendage yields

$$\int_{v_F} \delta\xi_F \cdot \left(\mathbf{f} - \rho\ddot{\xi}_F\right) dV = \int_{v_F} \delta\epsilon : \sigma \, dV \tag{6.15}$$

with

$$\xi_F = \mathbf{R} + \mathbf{A} + r_F + \eta \equiv \mathcal{F}_I^T \underline{R} + \mathcal{F}_b^T \underline{A} + \mathcal{F}_k^T (\underline{r}_F + \underline{\eta})$$

$$\ddot{\xi}_F = \mathcal{F}_b^T \left[\dot{\underline{u}} + \underline{\omega}_b \times \underline{u} + \dot{\underline{\omega}}_b \times \underline{A} + \underline{\omega}_b \times \left(\underline{\omega}_b \times \underline{A}\right)\right]$$

$$+ \mathcal{F}_k^T \left[\dot{\underline{\omega}}_k \times (\underline{r}_F + \underline{\eta}) + 2\,\underline{\omega}_k \times \dot{\underline{\eta}} + \underline{\omega}_k \times \left(\underline{\omega}_k \times (\underline{r}_F + \underline{\eta})\right) + \ddot{\underline{\eta}}\right] \equiv \mathcal{F}_b^T \underline{a}_b \equiv \mathcal{F}_k^T \underline{a}_k$$

$$\delta\xi_F = \mathcal{F}_b^T \left[\delta\underline{x} + \delta\underline{\theta}_b \times \underline{A}\right] + \mathcal{F}_k^T \left[\delta\underline{\theta}_k \times (\underline{r}_F + \underline{\eta}) + \delta\underline{\eta}\right]$$

where

$\underline{u}$	=	velocity of body frame wrt inertial frame
$\underline{\omega}_b$	=	angular rate of body frame wrt inertial frame
$\underline{\omega}_k$	=	angular rate of appendage frame wrt inertial frame
$\delta\underline{x}$	=	virtual displacement of the body frame
$\delta\underline{\theta}_b$	=	virtual rotation of the body frame
$\delta\underline{\theta}_k$	=	virtual rotation of the appendage frame
$\delta\underline{\eta}$	=	virtual relative displacement

The appendage angular rate can be decomposed into

$$\omega_k = \mathcal{F}_k^T \underline{\omega}_k = \omega_b + \omega_{bk} = \mathcal{F}_b^T \underline{\omega}_b + \mathcal{F}_k^T \underline{\omega}_{bk} \tag{6.16}$$

Similarly,

$$\delta\theta_k = \mathcal{F}_k^T \delta\underline{\theta}_k = \mathcal{F}_b^T \delta\underline{\theta}_b + \mathcal{F}_k^T \delta\underline{\theta}_{bk} \tag{6.17}$$

where

$\underline{\omega}_{bk}$	=	relative angular rate of appendage frame wrt body frame
$\delta\underline{\theta}_{bk}$	=	relative virtual rotation of the appendage frame wrt body frame

The frames are transformed with

$$\mathcal{F}_b = C_{bk}\,\mathcal{F}_k \tag{6.18}$$

Corresponding to Equation (6.9), the rigid degrees of freedom define

$$\{\ddot{u}_R\} = \begin{Bmatrix} \ddot{\underline{u}} \\ \dot{\underline{\omega}}_b \\ \dot{\underline{\omega}}_{bk} \end{Bmatrix} \tag{6.19}$$

Substitution into Equation (6.15) yields following relations.

$$\begin{Bmatrix} \left(\int_{v_F} C_{bk}\,\ddot{\underline{\eta}}\,\rho\,dV\right) \\ \left(\int_{v_F} \left[\underline{A}\times\left(C_{bk}\,\ddot{\underline{\eta}}\right) + C_{bk}\left((\underline{r}_F+\underline{\eta})\times\ddot{\underline{\eta}}\right)\right]\rho\,dV\right) \\ \left(\int_{v_F} (\underline{r}_F+\underline{\eta})\times\ddot{\underline{\eta}}\,\rho\,dV\right) \end{Bmatrix} \rightarrow [B]\,\{\ddot{u}_F\} \tag{6.20}$$

$$\begin{Bmatrix} \left(\int_{v_F} C_{bk}\,\underline{f}_F\,dV\right) \\ \left(\int_{v_F} \left(\underline{A}+C_{bk}\,(\underline{r}_F+\underline{\eta})\right)\times\left(C_{bk}\,\underline{f}_F\right)dV\right) \\ \left(\int_{v_F} (\underline{r}_F+\underline{\eta})\times\underline{f}_F\,dV\right) \end{Bmatrix}$$

$$\begin{Bmatrix} \left(2\int_{v_F} C_{bk}\left(\underline{\omega}_k\times\dot{\underline{\eta}}\right)\rho\,dV\right) \\ \left(2\int_{v_F} \underline{A}\times\left(C_{bk}\,(\underline{\omega}_k\times\dot{\underline{\eta}})\right) + C_{bk}\left((\underline{r}_F+\underline{\eta})\times(\underline{\omega}_k\times\dot{\underline{\eta}})\right)\rho\,dV\right) \\ \left(2\int_{v_F} \left((\underline{r}_F+\underline{\eta})\times(\underline{\omega}_k\times\dot{\underline{\eta}})\right)\rho\,dV\right) \end{Bmatrix} \rightarrow \{R_{RF}\} \tag{6.21}$$

The equations for the flexible domain results in

$$\int_{V_F} \delta\underline{\eta}^T \underline{f}_F \, dV = \int_{V_F} \delta\underline{\eta}^T \underline{a}_k \, \rho \, dV + \int_{V_F} \delta\varepsilon : \sigma \, dV \tag{6.22}$$

Equations (6.20) - (6.22) yield necessary equations to generate the full equations of motion for a vehicle with articulated flexible appendage. All terms are retained. Vehicle specific truncation of nonlinear terms should be made to these equations. Detailed derivation using an alternate approach of a vehicle with articulated flexible appendage is provided in [5].

Validation

To validate the concept of mating flexible domain equations with code generated using a symbolic manipulation software, the HS-601 satellite [5] is chosen for implementation. AUTOLEV software generated the rigid body portion of the code. Rest of the code that deals with flexible domain has to be coded by the user.

The specific example applies 40 ft-lbs of torque about the roll-axis for .1 sec. The magnitude of the torque characterizes the authority of the reaction control thrusters. The angular positions and rates are shown in Figures 6.3 - 6.4. The results are identical to the previous simulation that was implemented using the derivation described in [5]. Note that the response shown characterizes an actual vehicle. The effects of the flexibility are pronounced. In the rate plots, the magnitude of the rates peak at about 4 times the rate expected for a rigid vehicle with same applied torque.

The use of a symbolic manipulation software with flexible domain equations offers large time saving in terms of both derivation and implementation. In this context, this approach is practical as an engineering tool.

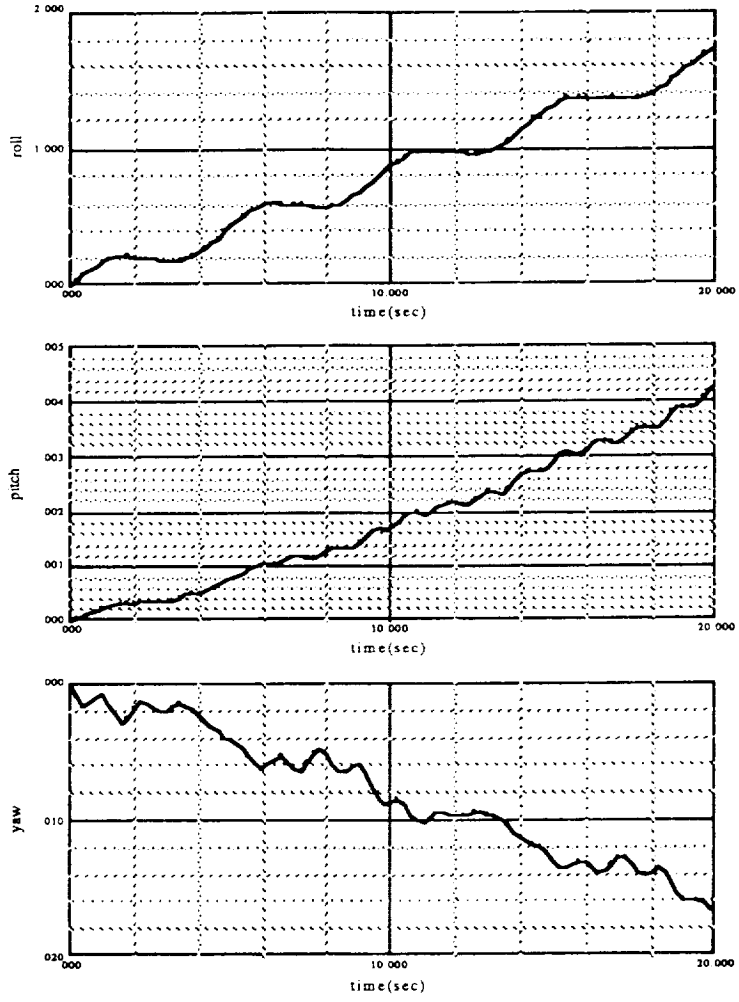


Figure 6.3. Angular Position (deg).

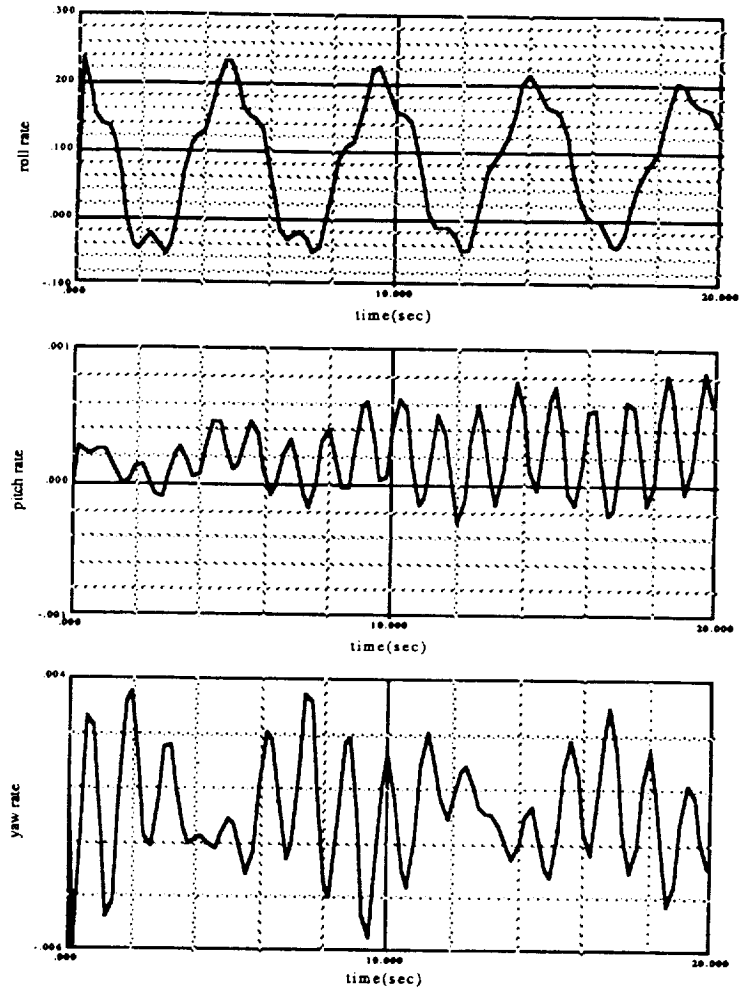


Figure 6.4. Angular Rate (deg/sec).

7. Simulation Performance

Digital simulations employing the hybrid dynamic model with the complete, nonlinear body-stabilized dynamics have been performed to demonstrate the stationkeeping control performance. Key simulation parameters which were added to produce worst case transient errors are: (1) 1.0 inches spacecraft CG offset from the pressure center of the south maneuver thrusters along the z-axis, (2) 5 lbf thrusters with 5% thrust mismatch producing the worst case acceleration disturbance, (3) 1.0 deg thruster misalignment in the direction adding disturbance, (4) flexible north and south wings at 0 deg wing angle, (5) 2 RTI control transport delay, (6) solar wing drive with ±0.25 deg deadband, (7) thruster EPW (Electrical Impulse Width) error model, which computes the thruster impulse on time delay as a function of time since last pulse. To accommodate the control loss due to EPW error, a fixed 4 msec thruster delay compensation was added to the command pulsewidth, (8) 8 msec pulsing constraint, (9) momentum wheel spinning at 45 ft-lb-sec throughout maneuver, (10) 0.035 deg sample to sample three sigma earth sensor noise, and (11) gyro sensor noises: rate random walk at PSD = $(10^{-7}\ d/s^2)^2/Hz$, angle random walk at PSD = $(10^{-4}\ d/s^2)^2/Hz$, angle noise at PSD = $(1.3 \times 10^{-5}\ d/s^2)^2/Hz$ and quantization of 0.3 arcsec.

With nominal structural mode frequencies and 0.5% structural damping, Figure 7.1 shows the Stationkeeping Mode control performance from a 100-sec south maneuver. The spacecraft angular position and rate along with their estimates about each control axis are plotted. It also shows the acceleration bias estimate and the control acceleration command about each axis. The acceleration bias estimates in this run were initialized to zero. The roll transient, which was induced primarily by the spacecraft yaw CG offset and the thrust mismatch, reaches 0.09 deg, while the yaw transient was primarily due to the thrust mismatch and was about 0.045 deg. The pitch transient induced by the combined effect of canted and thrust mismatch is 0.01 deg. In the steady state, a limit cycle about the pitch axis resulted from the 8 msec thrusting constraint. The acceleration bias estimation converges within 5 sec, showing a smooth bias estimate in the steady state. Transient errors can be improved with an initialization of the acceleration bias estimates to their steady state values. Figure 7.2 shows the performance with both the roll and yaw acceleration bias estimates initialized to 4.5% off their steady state values. Due to an over estimate of the thrusting bias by 4.5%, the roll and yaw attitude were over controlled, yielding transients up to 0.06 deg and 0.042 deg, respectively, in the direction opposite to Figure 7.1. Ideally, a perfect initialization would result in an significant reduction to the roll and yaw transients.

Effects of structural mode uncertainty and damping were also investigated through simulations. Figure 7.3 shows the performance with a 100% frequency increase to each mode, and Figure 7.4 a 50% frequency drop. In both cases, a 0.25% structural damping was assumed. A 50% (Δ) frequency error is equivalent to a 75% [$= 1-(1-\Delta)^2$] change to stiffness of the flexible wings. It is not expected that the structural modes at the on orbit deployed condition would exceed more than 20% from the nominals. Likewise, the structural damping nominally exceeds 0.5%. The modal errors given above are to demonstrate the robustness of the control system. The simulation results show that variations in modal parameters have no major effects on the modal stability nor degrade the pointing performance.

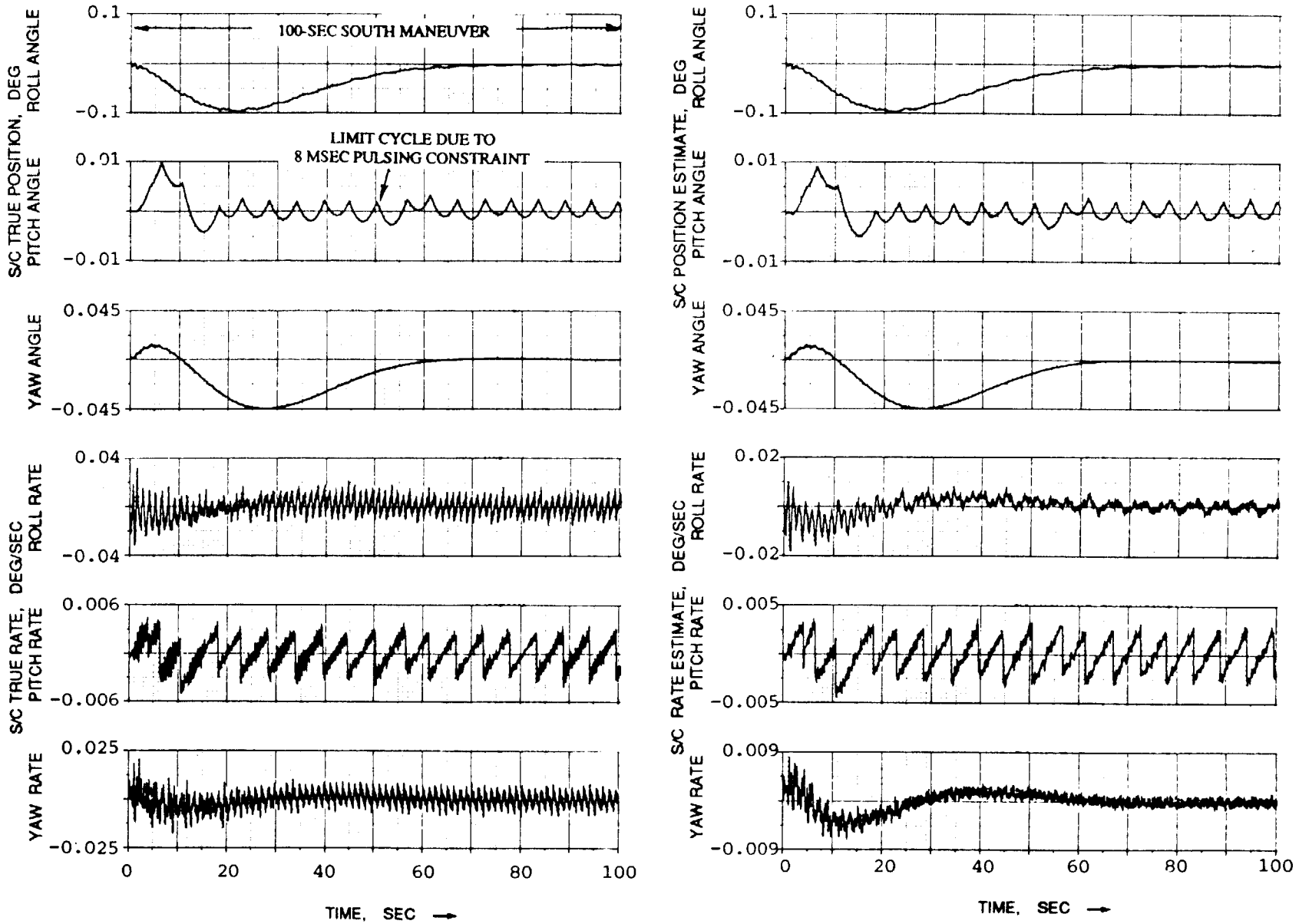


Figure 7.1 Stationkeeping Mode Control Performance From A 100 -sec South Maneuver with Zero Initial Acceleration Bias Estimates - Nominal Modal Frequencies, 0.5% Structural Damping

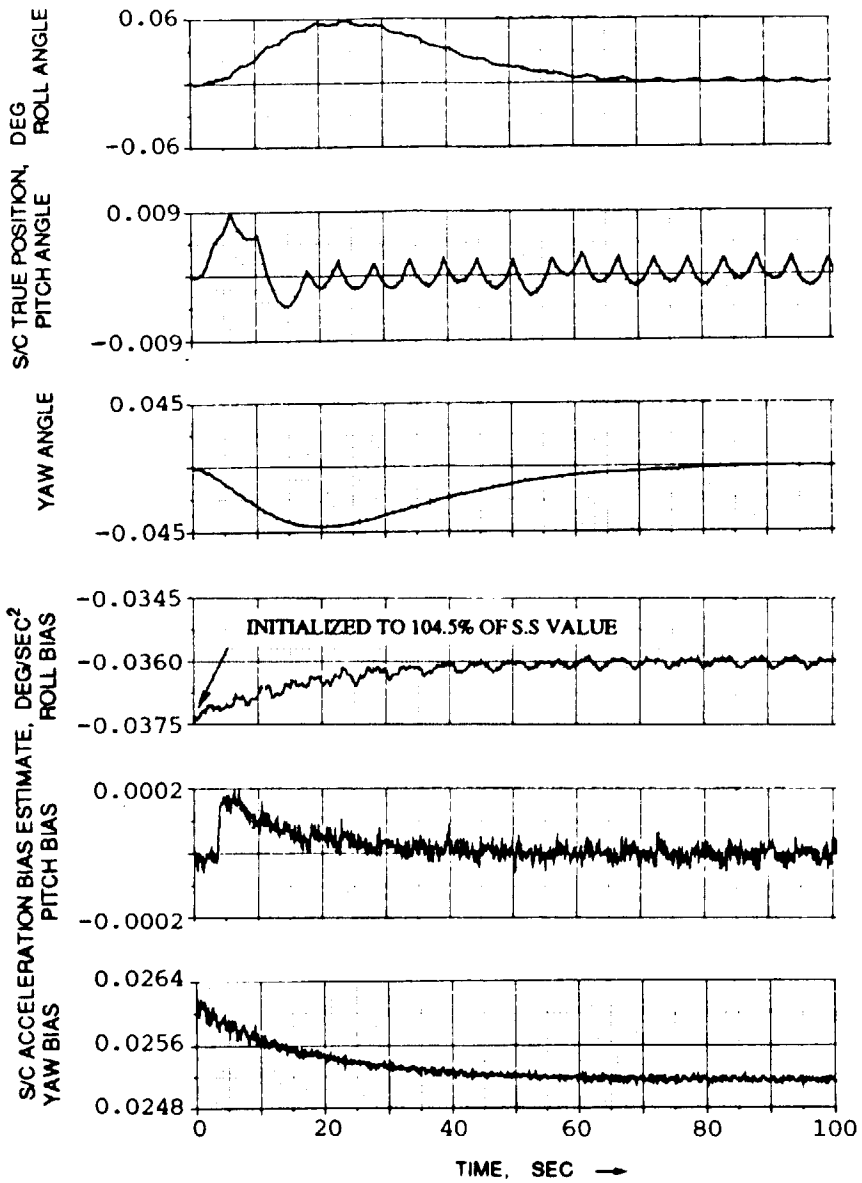


Figure 7.2 Stationkeeping Mode Control Performance From A 100 -sec South Maneuver with Acceleration Bias Estimates Initialized to 4.5% off Steady State Values - Nominal Modal Frequencies, 0.5% Structural Damping

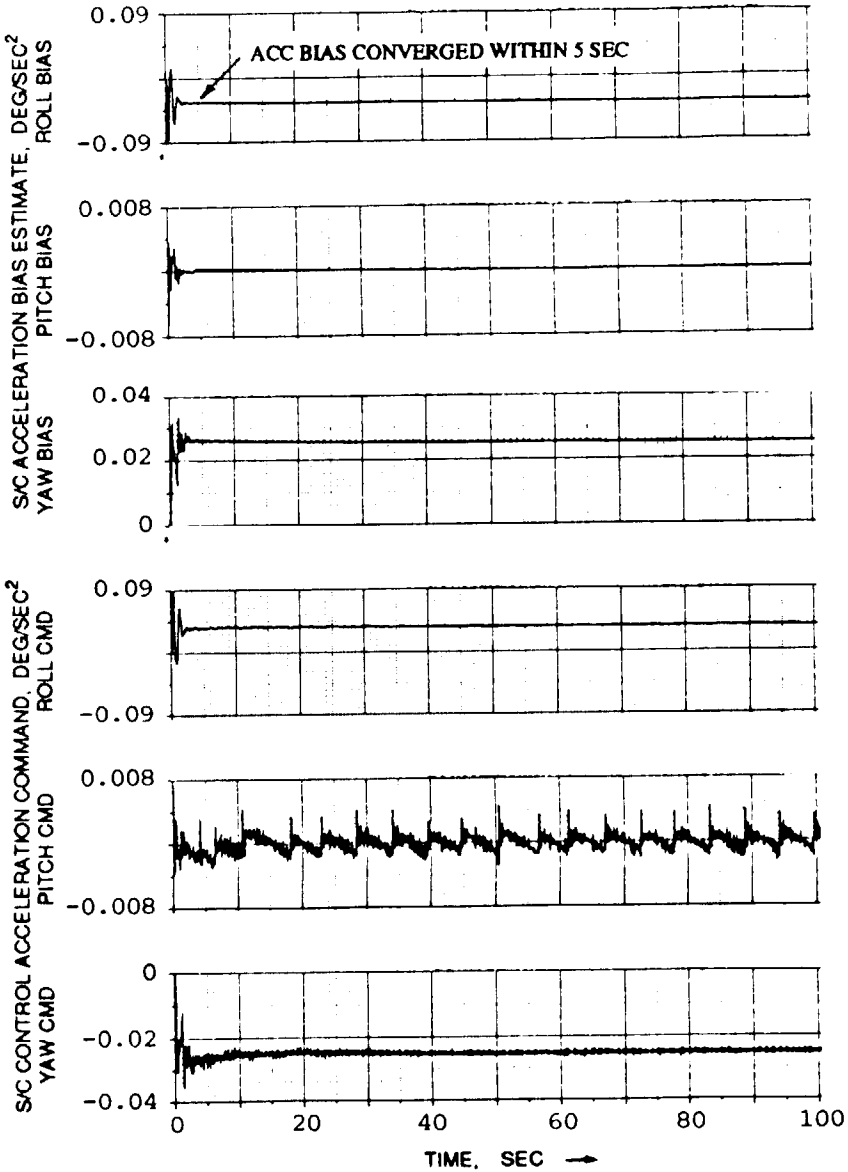


Figure 7.1 (continued)

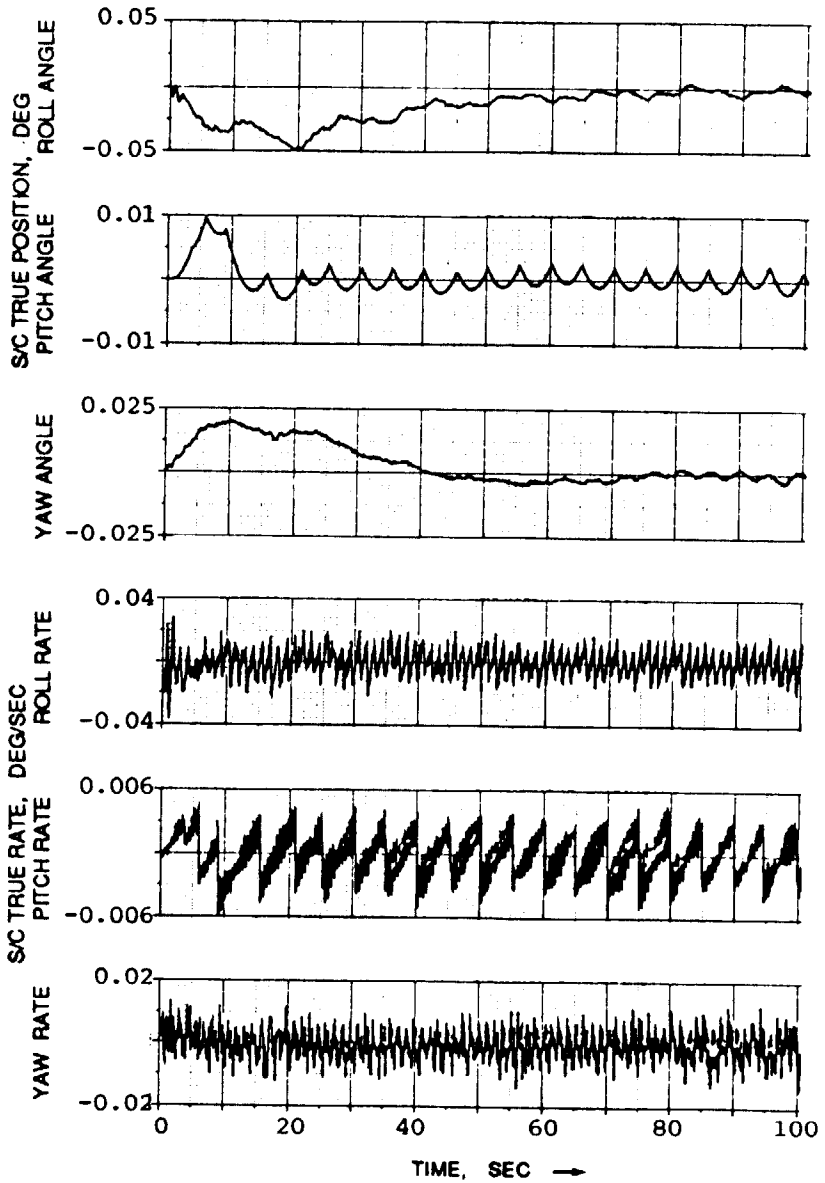


Figure 7.4 Stationkeeping Mode Control Performance From A 100 -sec South Maneuver with Zero Initial Acceleration Bias Estimates - 50% Modal Frequency drop, 0.25% Structural Damping

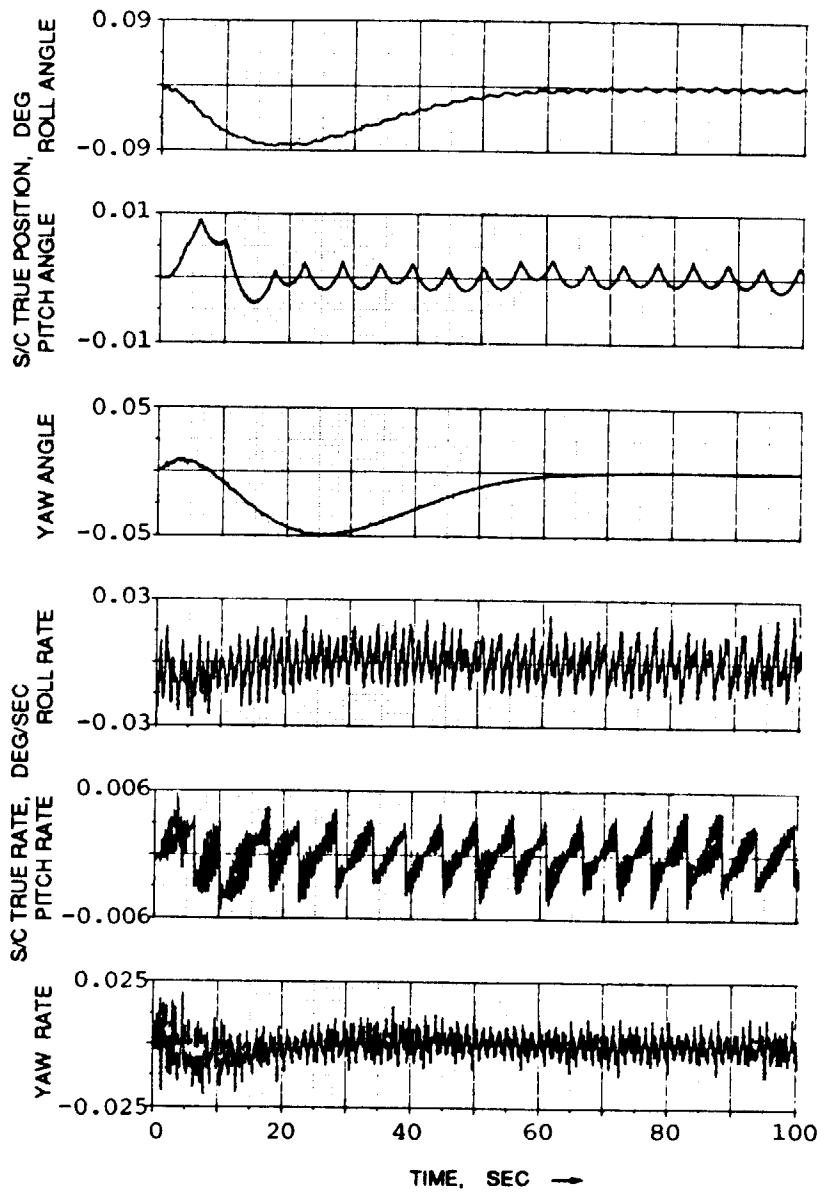


Figure 7.3 Stationkeeping Mode Control Performance From A 100 -sec South Maneuver with Zero Initial Acceleration Bias Estimates - 100% Modal Frequency Increase, 0.25% Structural Damping

8. Concluding Remarks

The design and analysis of a stationkeeping control system for a body-stabilized spacecraft having flexible solar wings of 3 solar panels per wing were presented. The use of the hybrid coordinate modeling approach along with frequency domain analysis technique accurately modeled the rigid-flex coupling behavior. The design philosophy to stabilize the structural modes and to smooth the flexure was discussed. The control system was designed to gain and/or phase-stabilize the structural modes. The lead inherent rate gyro references, the structural filters and the time-varying bias estimation gains were key factors to achieve a successful design. Control performance of Stationkeeping/Transition Modes during a south maneuver under the worst case simulation environment was demonstrated through digital simulation. The accuracy of the analytical model for structural mode/control loop interaction is best verified with the test data obtained from comprehensive ground testing. The control system as presented allows for a high degree of uncertainty on mode shape and frequency.

An alternate, indirect implementation of flexible vehicle dynamics has been presented. The effort required to derive and to implement the equations of motion can be significantly reduced. A formulation through the virtual work principle allowed consistent derivation and discretization of the flexible domain within the context of the finite element method. Extensions into nonlinear flexible models can be made.

Acknowledgement - A portion of this work supported by the C. S. Draper Laboratory Internal Research & Development is gratefully acknowledged.

References

1. P.C. Hughes, *Spacecraft Attitude Dynamics*, John Wiley & Sons (1986).

2. P.W. Likins, "Analytical Dynamics and Nonrigid Spacecraft Simulations", *Technical Report 32-1593*, Jet Propulsion Laboratory, Pasadena, CA (1974).

3. P.W. Likins and G.E. Fleischer, "Results of Flexible Spacecraft Attitude Control Studies Utilizing Hybrid Coordinates", *Journal of Spacecraft and Rockets*, Vol. 8, 264-273 (1971).

4. J. Storch and S. Gates, "Equations of Motion for a Flexible Spacecraft - Lumped Parameter Idealization", *Technical Report CSDL-R-1582*, C.S. Draper Laboratory, Cambridge, MA (1982).

5. S. Gates and D.S. Kang, "An Attitude Dynamics Model of the Hughes HS-601 Spacecraft", *Technical Report CSDL-R-2156*, C.S. Draper Laboratory, Cambridge, MA (1989).

6. *SD/FAST User's Manual, Version 88/02/20*, Symbolic Dynamics, Inc., Mountain View, CA (1988).

7. D.E. Rosenthal and M.A. Sherman, "High Performance Multibody Simulations via Symbolic Equation Manipulation and Kane's Method", *Journal of the Astronautical Sciences*, Vol. 34, No. 3, 223-239 (1986).

8. D.B. Schaechter, D.A. Levinson, T.R. Kane, *AUTOLEV User's Manual*, OnLine Dynamics, Inc., Sunnyvale, CA (1988).

9. M. Sayers, *AUTOSIM User's Manual, Version 1.0 B8*, Univ. of Michigan, Transportation Research Institute, Ann Arbor, MI (1990).

10. B. Wie and C.T. Plescia, "Attitude Stabilization of Flexible Spacecraft During Stationkeeping Maneuvers", *Journal of Guidance and Control, AIAA* (1983).

11. A.N. Penchuk and P.D. Hattis, "A Frequency Domain Stability Analysis of a Phase Plane Control System", *J.Guidance*, Vol. 8, No. 1 (1985).

12. J.F. Yocum and L.I. Slafer, "Control System Design in the Presence of Severe Structural Dynamics Interactions", *J.Guidance*, Vol. 1, 109-116 (1978).

13. L.I. Slafer, "In-Orbit Evaluation of the Control System/Structural Mode Interactions of the OSO-8 Spacecraft", *Dynamics and Control of Large Flexible Spacecraft Proceedings of the Second VPI&AIAA Symposium*, June 21-23, 111-127 (1979).

14 D.S. Kang, "Present Finite Element Technology from a Hybrid Formulation Perspective", *Computers and Structures* , 35, 321-327 (1990).

TRANSFORM METHODS FOR PRECISION CONTINUUM AND CONTROL MODELS OF FLEXIBLE SPACE STRUCTURES

Victor D. Lupi, Research Assistant
Massachusetts Institute of Technology
Cambridge, Massachusetts

James D. Turner,* Controls and Dynamics Group Leader
Hon M. Chun,* Staff Scientist
Photon Research Associates, Inc.
Cambridge, Massachusetts

ABSTRACT

An open loop optimal control algorithm is developed for general flexible structures, based on Laplace transform methods. A distributed parameter model of the structure is first presented, followed by a derivation of the optimal control algorithm. The control inputs are expressed in terms of their Fourier series expansions, so that a numerical solution can be easily obtained. The algorithm deals directly with the transcendental transfer functions from control inputs to outputs of interest, and structural deformation penalties, as well as penalties on control effort, are included in the formulation. The algorithm is applied to several structures of increasing complexity to demonstrate its generality.

1. INTRODUCTION

The control of large flexible structures has become an important issue in recent years, primarily in the aerospace industry.[1] As larger structures continue to be deployed in space, the effects of control-structure interaction are becoming increasingly important. For example, stringent pointing requirements for space-based antennae make it necessary to isolate and suppress unwanted structural vibration caused by both slewing maneuvers and exogenous disturbances. Consequently, it becomes necessary to model structural flexibility when developing control laws for these types of structures.

Because disturbances and control forces generally act at discrete points on the structure, structural responses tend to exhibit wave propagation characteristics. Traditional finite element codes are unable to capture the high frequency behavior of such structures, due primarily to the spatial discretization associated with lumped parameter models. This limitation makes it particularly difficult to study the propagation of flexural waves within structures, since an extremely fine discretization is required to preserve the local wave-like characteristics of the disturbances. To overcome this problem, this paper develops a distributed parameter, system-based model, which deals directly with the governing partial differential equations that describe the structure.

Given the continuum model of a flexible structure, there remains the issue of identifying control methodologies that take advantage of the additional high frequency information available therein. Tzafestas[2] develops a distributed parameter analogue of the linear quadratic regulator theory. A distributed parameter Riccati equation, expressed in terms of spatial differential operators, is presented. Miller, Hall, and von Flotow[3] develop optimal control laws for power flow at structural junctions based on a travelling wave approach. The effect of the localized controller is to modify the wave scattering matrix at the junction in a way that minimizes the power flowing from the junction. MacMartin and Hall[4] consider optimal control of power flow in uncertain structures based on an H_∞ cost criterion. Closed-loop stability is guaranteed by minimizing the maximum power imparted to the structure over all frequencies. The optimal distributed control of a rigid spacecraft with flexible appendages is discussed by Breakwell.[5]

Skaar[6] presents closed-form open loop optimal control solutions for a simple structure. The cost function considered has the form:

$$J = \int_0^{t_f} \left\{ k_1 u(t)^2 + k_2 \dot{u}(t)^2 \right\} dt \qquad (1)$$

where u represents the control input, and k_1 and k_2 are constants. Terminal and integrated penalties on the structural deformations are not permitted. Rather, the terminal constraints are adjoined to the cost function with Lagrange multipliers. The exclusion of deformational penalties makes it possible to derive analytical solutions for certain types of maneuvers. Otherwise, the optimal control solution can, in general, only be obtained by numerical methods.

Analytical results are available for only the simplest of distributed parameter models, containing very few flexible elements. More often, a complex structure, such as a truss beam, is replaced by a single equivalent member in the continuum model. Such an approximation is usually accurate at low frequencies only. For general structures, the structural responses must be calculated numerically. The convolution integral representation technique developed by Skaar is generalized in this paper to handle arbitrary structural configurations.

A review of the continuum modeling approach is presented in section 2. The optimal control formulation is developed in section 3. Several examples of this method, applied to structures of increasing complexity, are then presented in section 4. Conclusions and recommendations can be found in section 5.

2. STRUCTURAL MODELING

2.1 Modeling of Flexible Elements

Traditional approaches for modeling complex structures have relied on finite element modeling techniques. This approach idealizes a structure as an assembly of many small pieces which are constrained to move together in a manner consistent with the internal elastic behavior of the underlying continuum model. These techniques are powerful and widely used. However, they suffer from various modeling idealizations which limit the accuracy of behavior predictions, particularly for high frequency.

In order to better model the high frequency behavior of

*Member, AIAA

elastic frame-like structures, a continuum approach is presented in this paper which overcomes the conventional limitations of traditional finite element modeling techniques. The continuum method (also known as the exact finite element, distributed parameter, or dynamic stiffness method), deals directly with the governing partial differential equations for the individual elastic elements to eliminate the explicit time dependence in the equations of motion. The Laplace transform is employed to convert the governing partial differential equations into ordinary differential equations in the spatial dimension. For common element models (e.g., rods in torsion, Bernoulli-Euler beams in bending), simple analytical solutions to these equations exist. The resulting solutions are explicit functions of the generally complex frequency parameter, s, which has been introduced through the application of the Laplace transform technique.

As an example, consider the case of a rod in torsion, shown in Fig. 1. The governing partial differential equation is

$$GJ\theta''(x,t) + mr^2\ddot{\theta}(x,t) = \tau(x,t) \tag{2}$$

where θ is the cross sectional angle of twist, GJ is the torsional rigidity, m is the mass per unit length, r is the cross sectional radius of gyration, $(\,)'=\partial(\,)/\partial x$, $\dot{(\,)}=\partial(\,)/\partial t$, and τ is the distributed torque. Applying the Laplace transform, we obtain

$$\theta''(x,s) + \frac{mr^2s^2}{GJ}\theta(x,s) = 0 \tag{3}$$

where initial conditions and distributed forcing along the element have been temporarily neglected. The general solution follows as:

$$\theta(x,s) = A(s)\cos\beta x + B(s)\sin\beta x, \quad \beta^2 = -\frac{mr^2}{GJ}s^2 \tag{4}$$

where the functions A(s) and B(s) depend on the constraints imposed at the boundaries of the element. To simplify the mathematical developments, a structural state vector is defined for the elastic element as follows:

$$y(x,s) = \begin{bmatrix} \theta(x,s) \\ T(x,s) \end{bmatrix} \tag{5}$$

where T represents the net torque resultant along the rod. With knowledge of the state at one boundary of the element, it is then possible to determine the state at any internal location. A frequency-dependant spatial transition matrix is used to propagate the system state to arbitrary element locations. Its transcendental elements can be thought of as spatially varying transfer functions. For this example, the transition matrix can be shown to be

$$\Phi(x,s) = \begin{bmatrix} \cos\beta x & \frac{1}{GJ\beta}\sin\beta x \\ -GJ\beta\sin\beta x & \cos\beta x \end{bmatrix} \tag{6}$$

and

$$y(x,s) = \Phi(x,s)\, y(0,s) \tag{7}$$

Alternatively, when the displacements at the boundaries are known, they can be related to the forces. This is accomplished by a dynamic stiffness matrix. Its name derives from the stiffness matrix associated with the traditional finite element method. For the rod in torsion, this matrix is given by

$$K(s) = \frac{GJ\beta}{\sin\beta l}\begin{bmatrix} \cos\beta l & -1 \\ -1 & \cos\beta l \end{bmatrix} \tag{8}$$

where l is the length of the rod.

In addition to torsional rods, the formulation also handles Euler beams in bending in two directions and axial rods. Timoshenko beams can also be included as continuum elements.

Because the continuum method treats each elastic member as a single element, no spatial discretization is required. This is in marked contrast with traditional finite element methods, where each element must be lumped into several segments. As a result, the continuum model is valid at all frequencies, insofar as the partial differential equation represents the actual physical structure. In contrast, the spatial discretization associated with the finite element approach creates a computational burden for even the simplest of structures (each flexible element is typically broken down into more than ten segments). This significant reduction in the number of discrete modeling elements required makes the continuum method more attractive from a computational point of view.

Another advantage to the frequency domain modeling approach is the ease with which damping is incorporated into the structural model. For internal damping (where energy is dissipated as heat within the structural elements) the static bending, axial or torsional stiffness is replaced by a complex valued function of the complex frequency. The functional relationship depends on the type of damping modeled. For example, a fractional derivative damping model scales the static stiffness by the square root of the complex frequency.[7] This type of damping model is extremely difficult to implement in time domain formulations, and requires a knowledge of the entire past history of the deformation of the structural element. For external damping (where structural energy is dissipated to the surroundings) the mass per unit length is replaced by a frequency-dependant parameter.[7]

2.2 Assembly of Elements

The assembly of flexible elements into a complex frame-like structure is accomplished using the method of local/global coordinates,[7] which is implemented in most finite element software. The structure to be modeled is divided into a set of flexible elements and a set of rigid joints, which attach to any number of flexible elements at their respective boundaries, as shown in Fig. 2. External forces are applied at the joints only, but the deformation of the structure is available at all points. (The case of a concentrated force applied within a

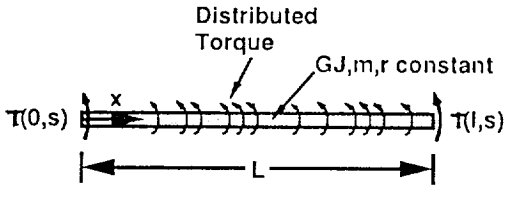


Fig. 1: Example of a distributed parameter element: a rod in torsion.

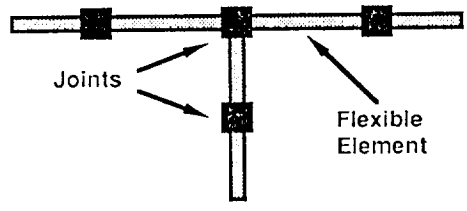


Fig. 2: Generic frame-like structure.

flexible element is treated by breaking the member into two continuous elements connected by a massless joint. Modeling distributed forcing is somewhat more difficult.) Massive joints are modeled by dynamic stiffness matrices as well, so that their contribution to the structural response at all frequencies is retained.

The topology of the structure is given by a connectivity matrix, which relates the local displacements of the structural elements (both flexible and rigid) to a set of global displacements which uniquely describe the location and orientation of all joints in the structure. The applied forces at the joints are defined in a dual manner, so that for every global displacement there exists a global forcing input at the same point and in the same direction. It is then a simple matter to compute the dynamic stiffness matrix for the structure as a whole at any given complex frequency. The individual stiffness matrices are first arranged in a large block diagonal matrix. This matrix is then post- and pre-multiplied by the connectivity matrix and its transpose, respectively, resulting in the system dynamic stiffness matrix.

Because the dynamic stiffness matrix is transcendental in nature, computing natural frequencies is not a simple matter of solving an eigenvalue problem, as is the case for the finite element approach. Rather, the stiffness matrix must be computed at many frequencies in order to gradually converge on each modal frequency. However, a powerful algorithm is available which rapidly converges on these eigenfrequencies.[8] The algorithm works for undamped systems only, and additional root-searching algorithms must be employed when structural damping is modeled.[7]

The system dynamic stiffness matrix can be inverted, if desired, yielding the dynamic flexibility matrix for the structure. Its elements can be thought of as transfer functions from the joint forces to the joint displacements or as admittance functions. This flexibility matrix, in conjunction with the flexible element spatial transition matrices, enables the straightforward calculation of the transfer function from any joint force to any point on the structure.

2.3 Inverse Laplace Transform

In the frequency domain, the responses at various locations within a linear elastic structure to multiple control inputs are determined by exploiting the principle of superposition. The set of responses, $y_i(s)$, are expressed as

$$y_i(s) = \sum_{j=1}^{m} g_{ij}(s)\, u_j(s), \quad i = 1,\ldots,r \tag{9}$$

where $g_{ij}(s)$ is the transfer function from the j'th input, $u_j(s)$, to the i'th output, $y_i(s)$. These equations can be expressed in more compact form using matrix notation:

$$\mathbf{y}(s) = \mathbf{G}(s)\mathbf{u}(s) \tag{10}$$

The matrix $\mathbf{G}(s)$ is the dynamic flexibility matrix for the given structure, or some partition of it, depending on the inputs and outputs considered.

The time response vector corresponding to $\mathbf{y}(s)$ is available via the inverse Laplace transform, given by

$$\mathbf{y}(t) = \frac{1}{2\pi j} \int_{\alpha-j\infty}^{\alpha+j\infty} \mathbf{y}(s)\, e^{st}\, ds \tag{11}$$

where the integration path is known as the Bromwich contour.[9] For finite dimensional systems, a residual expansion is used in lieu of Eq. (11) to compute the time responses analytically. The same can be done for distributed parameter systems, except that the expansion has an infinite number of terms and must therefore be truncated at some point. However, greater numerical accuracy is possible by working with Eq. (11) directly. The frequency domain response is tabulated for values of complex frequency equally spaced along the Bromwich contour, and a numerical procedure converts this data into a response history evaluated at equal spaces in time.[10]

Because this approach utilizes frequency domain representations of the control inputs, it circumvents the computationally expensive calculation of convolution integrals. Furthermore, signals that cannot be represented by finite dimensional state space models are easily handled in the frequency domain. For example, the implementation of a time delay simply requires multiplication of the frequency domain data by a suitable exponential of the complex frequency before the inverse Laplace transform algorithm is invoked. Implementing such a time delay on a modal basis requires a truncated series expansion of the complex exponential (such as a Padè approximation), with many terms needed to obtain an accurate representation.

3. OPTIMAL CONTROL FORMULATION

With the continuum modeling approach described above, it is possible to recast a class of optimal control problems into a convenient form, from which optimal control trajectories are easily calculated. This form is applicable to a completely general frame-like structure (although applying this method to structures containing plates and membranes is the subject of current research[11]), with multiple control inputs and multiple outputs. The class of problems discussed here are fixed-time, linear quadratic, open loop control problems with penalties on control effort, position and velocity of various output points on the structure, and structural deformation. Thus, the cost functional has the form

$$J = [\mathbf{y}(t_f) - \mathbf{y}_d]^T \mathbf{H} [\mathbf{y}(t_f) - \mathbf{y}_d] + \int_0^{t_f} \left\{ [\mathbf{y}(t_f) - \mathbf{y}_d]^T \mathbf{Q} [\mathbf{y}(t_f) - \mathbf{y}_d] + \mathbf{u}(t)^T \mathbf{R}\, \mathbf{u}(t) \right\} dt \tag{12}$$

where $\mathbf{H}$, $\mathbf{Q}$ and $\mathbf{R}$ are weighting matrices, and $\mathbf{y}_d$ is the vector of desired output values. It should be noted that $\mathbf{y}_d$ represents the physical output variables of interest, and is not related to the outputs of some state variable representation of the system.

Traditionally, the dynamics of the system are adjoined to this functional via a costate vector as differential equation constraints. However, because the structural transfer functions are transcendental and infinite dimensional, a finite dimensional costate vector cannot be defined. Tzafestas[2] succeeded in identifying a distributed parameter optimal control solution which incorporates an infinite dimensional costate. This solution represents the distributed parameter analogue of the Riccati differential equation for finite dimensional systems. However, the method is not immediately applicable to complex structures, where more than one partial differential equation is involved. Even the case of a single beam in bending presents considerable difficulty.[12]

One alternative to adjoining an infinite dimensional costate is modal truncation. The high frequency modes of the structure are simply ignored, and the dynamics of the structure is approximated with a finite dimensional state space realization. However, in order to take advantage of the "exactness" of the continuum modeling approach, it seems appropriate to avoid modal truncation altogether. Instead, we express each response as the convolution of impulse responses with control inputs. This yields for the cost function

$$J = \left[\int_0^{t_f} G(t_f-\tau)u(\tau)\,d\tau - y_d\right]^T H \left[\int_0^{t_f} G(t_f-\tau)u(\tau)\,d\tau - y_d\right] + \int_0^{t_f} \left\{ \left[\int_0^{t} G(t-\tau)u(\tau)\,d\tau - y_d\right]^T Q \left[\int_0^{t} G(t-\tau)u(\tau)\,d\tau - y_d\right] + u(t)^T R u(t) \right\} dt \quad (13)$$

where G(t) is the matrix of impulse responses from each control input to each output, as defined in section 2.3. These convolution integrals need not be computed directly, as they are the inverse transforms of the appropriate transfer functions multiplied by the associated control inputs. Taking variations in **u**, we are left with an integral equation which, except for a very small number of special cases, is difficult or impossible to solve in closed form. It is therefore necessary to express the control inputs as weighted sums of an appropriate set of basis functions which span the function space of allowable control inputs. For fixed time problems, the Fourier series is a good candidate, leading to

$$u(t) = \begin{bmatrix} f(t)^T c_1 \\ f(t)^T c_2 \\ \vdots \\ f(t)^T c_m \end{bmatrix} = F(t)^T c, \quad f(t) = \begin{bmatrix} 1 \\ \sin \pi t/t_f \\ \cos \pi t/t_f \\ \vdots \\ \sin n\pi t/t_f \\ \cos n\pi t/t_f \end{bmatrix} \quad (14)$$

where

$$F(t) = \begin{bmatrix} f(t) & & & \\ & f(t) & & \\ & & \ddots & \\ & & & f(t) \end{bmatrix}, \quad c = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix} \quad (15)$$

The cost functional now depends only upon a constant vector, **c**, representing the coefficients in the Fourier series expansions of the control inputs. The resulting cost functional is quadratic in the coefficient vector, and the minimization problem is straightforward, yielding:

$$c = A^{-1} B y_d \quad (16)$$

where

$$A = Y(t_f)^T H Y(t_f) + \int_0^{t_f} \left\{ Y(t)^T Q Y(t) + F(t)^T R F(t) \right\} dt \quad (17)$$

$$B = Y(t_f)^T H + \int_0^{t_f} \left\{ Y(t)^T Q \right\} dt \quad (18)$$

and

$$Y(t) = \int_0^{t} G(t-\tau) F(\tau)\,d\tau \quad (19)$$

Once again, the convolution of impulse responses with basis function inputs can be calculated via the inverse Laplace transform:

$$Y(t) = L^{-1}\{G(s)F(s)\} \quad (20)$$

Furthermore, these basis responses can be computed *a priori*, provided that the forcing locations and structural deformation penalty locations are known in advance. This makes it possible to try a large number of cost functionals without repeatedly calculating the responses to basis inputs. It should be mentioned that a large amount of memory is required to store this data.

A unique advantage of this approach is that it readily accommodates penalties in higher derivatives of both control effort and physical deformation. In the frequency domain, differentiation merely requires multiplication of the data by the Laplace transform variable. The inverse transformation will then produce the derivative of the original signal. Higher order derivatives are obtained by multiplying by higher powers of the complex frequency. Incorporating higher derivative penalties in the traditional optimal control formulation is considerably more difficult.

Special constraints on the control histories are treated by adjoining the constraints via Lagrange multipliers. An example is the requirement that the controls be continuous at the beginning and end of the maneuver. This implies that the controls go to zero at the initial and final times. Two Lagrange multipliers are therefore introduced for each control input. Performing the minimization, we obtain

$$\begin{bmatrix} c \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} A & F(0) & F(t_f) \\ F(0)^T & 0 & 0 \\ F(t_f)^T & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} B y_d \\ 0 \\ 0 \end{bmatrix} \quad (21)$$

where λ_1 and λ_2 are Lagrange multiplier vectors corresponding to the control constraints at the initial and final times, respectively.

It should be noted that the only approximation in the entire development involves expressing the control inputs in terms of the basis functions. The dynamics of the entire structure is accounted for, since the impulse responses are exact (insofar as the original equations represent physical reality). Also, the structural deformations are assumed to be small, so that linearization does not introduce significant errors. As a result, large angle slew maneuvers are not included in this class of problems. It is possible, however, to express structural deformations with respect to a nominal condition during a large angle slew, and then linearize about that reference.

It is important to note the difference between the approximations made in the continuum approach and those made in control systems based on finite element models. Typically, a finite element model is used to determine a truncated state space realization of the system. The control system is then designed using standard methodologies, such as linear quadratic regulator (LQR) or H_∞ theory. The modeling error associated with modal truncation must then be considered in assessing the performance of the system. In contrast to this, the continuum approach retains all modes of the system and therefore has no modeling error associated with it, provided that the original partial differential equations represent the physical system exactly. The only limitation to this approach is therefore the number of basis functions chosen to represent the control signals. Using the Fourier series expansion essentially places a limit on the bandwidth of the control inputs. If the bandwidth of the physical controls are known in advance, the number of basis functions used in the calculations can be adjusted accordingly. The resulting control history would then indeed be optimal for that set of physical controls.

4. EXAMPLES

4.1 Rigid Mass with Flexible Appendage

In an earlier analytical study by Skaar[6], the open loop control of a rigid mass with a flexible appendage, shown in Fig. 3, was studied. In his work, deformational penalties were not incorporated into the cost function; rather, the terminal conditions were adjoined to the cost functional as constraints. Skaar derived analytical expressions for impulse responses of the simple mass/appendage structure and thus obtained closed form optimal control solutions for the structure. His approach does not readily generalize for more complex structures. In contrast, the formulation presented here readily generalizes for realistic complex structures. Nevertheless, the mass/appendage structure is used as a first example to validate the optimal control formulation.

The maneuver involves translating the mass a distance of 10 meters along the axis of the flexible appendage, bringing it to rest with minimal residual energy and post-maneuver drift after 20 seconds. The first case places terminal penalties on the final position and velocity of the rigid mass and on a point 4/5 of the length along the flexible appendage. A small penalty is also placed on control rate, and 17 basis functions are used to approximate the control input. The results, shown in Fig. 4, indicate that the terminal conditions are matched, and residual energy is minimized. The second case places additional terminal penalties on points along the flexible appendage, as well as en-route (integrated) penalties on structural deformation at these points. As shown in Fig. 5, the magnitude of the structural deformation during the maneuver is reduced slightly (about 15%), and large deformation occurs over smaller intervals of time, resulting in a lower RMS displacement of the tip of the appendage over the maneuver interval. The terminal conditions are again matched. In the final case, the member stiffness is reduced by a factor of four, so that the primary modal frequency of the structure corresponds approximately

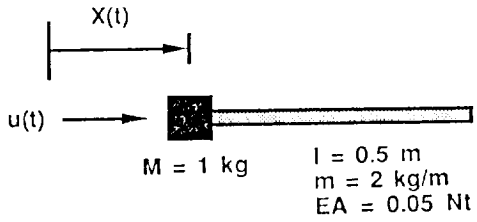
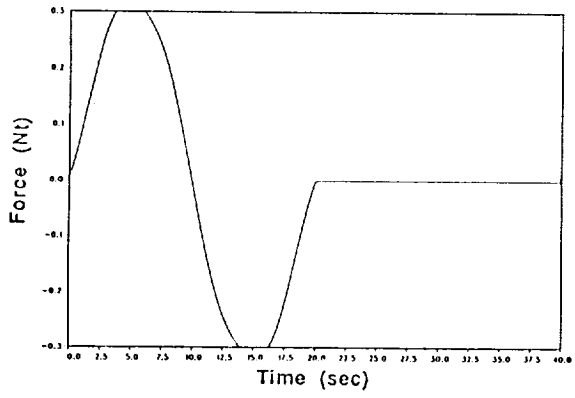
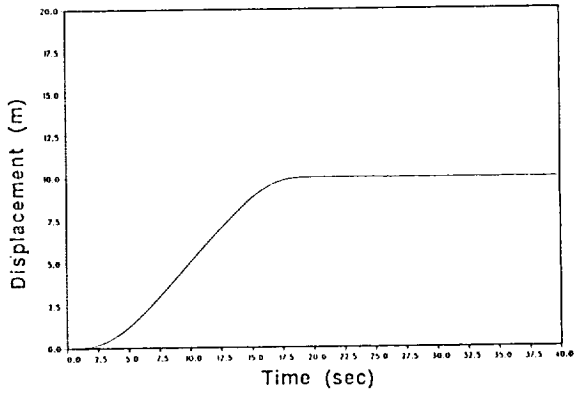


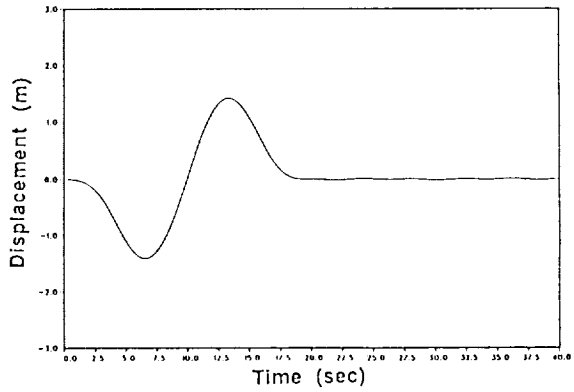
Fig. 3: Mass/flexible appendage system.



(a)



(b)



(c)

Fig. 4: Results of optimal maneuver of mass/flexible appendage system (case #1): (a) control force applied to rigid mass, (b) position of rigid mass, (c) deformation of tip of flexible appendage with respect to position of rigid mass.

with the frequency of the first basis function of the control input. The results of this case, presented in Fig. 6, indicate that the control input has been adjusted so that excitation of the primary mode of the structure is suppressed, at the expense of increased control effort. Again, the terminal conditions are matched, and residual internal energy is minimal.

4.2 Starfish Configuration

A typical model for a spacecraft with flexible appendages is the starfish configuration, shown in Fig. 7. It consists of a rigid hub, to which four flexible arms are attached. At the end of each arm is a rigid mass. Control inputs are available at the hub and at the tips of two of the flexible arms.

For this structural model, two maneuvers are presented. The first is an in-plane translation of 0.1 meter in the direction of one of the flexible appendages, with a final time of 5 seconds and all three control inputs available. Terminal penalties are placed on the position and velocity of the hub, as well as at three locations along the two arms whose axes lie in a direction perpendicular to the motion. It is anticipated that the control forces will excite bending vibration in these perpendicular members due to their inertia. Identical magnitude

and velocity penalties are also placed on the three control inputs. The results of this maneuver are shown in Fig. 8. The forces provided by the thrusters at the two appendages are identical (except for their sign) and differ slightly from the control history of the thruster at the hub. This discrepancy compensates for the inertia of the flexible arms, bringing the system to the final desired state with minimal residual energy.

The second maneuver involves a small (0.1 radian) counter-clockwise in-plane rotation about the hub in addition to the translation of the previous maneuver, in order to demonstrate the ability of the optimal control approach to handle multiple final conditions. Here, only the lateral forces at the tips of two of the appendages are available. Also, terminal position and velocity penalties are placed on two points along all four arms, as well as on the central hub, and the final time remains unchanged. The results are presented in Fig. 9. Again, the incorporation of state penalties involving structural deformation succeeds in minimizing residual internal energy at the terminal time. As previously mentioned, the linear control solution applied to this type of maneuver is only appropriate for small rotations and angular rates. For large rotations, nonlinear kinematics must be considered, whereas for large angular rates, gyroscopic forces become significant.

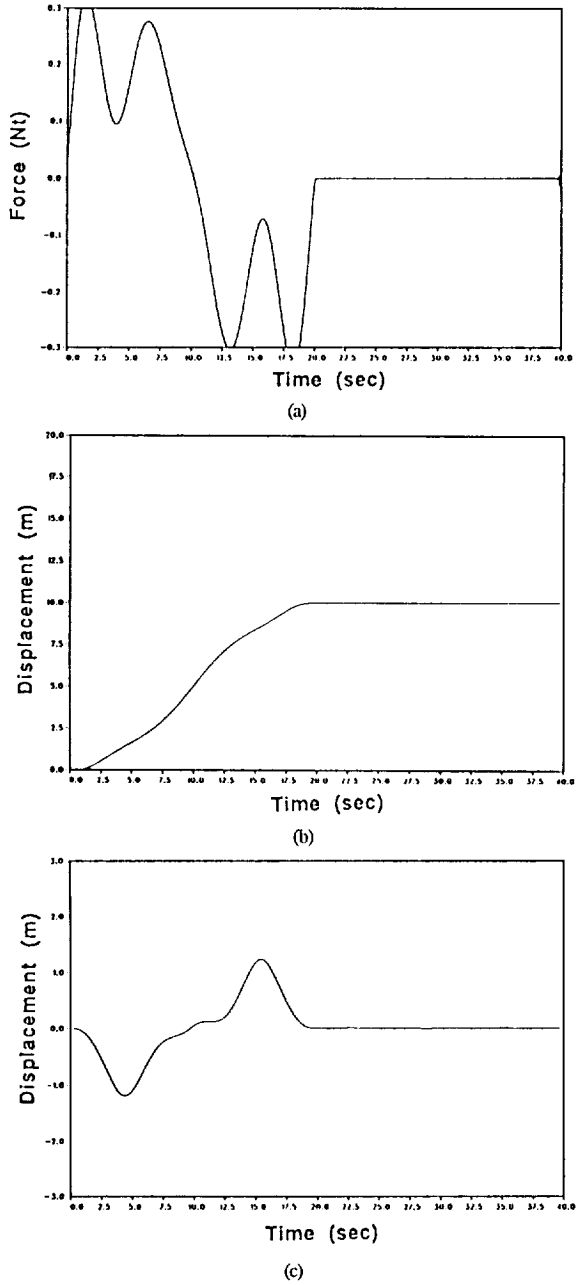


Fig. 5: Results of optimal maneuver of mass/flexible appendage system (case #2): (a) control force applied to rigid mass, (b) position of rigid mass, (c) deformation of tip of flexible appendage with respect to position of rigid mass.

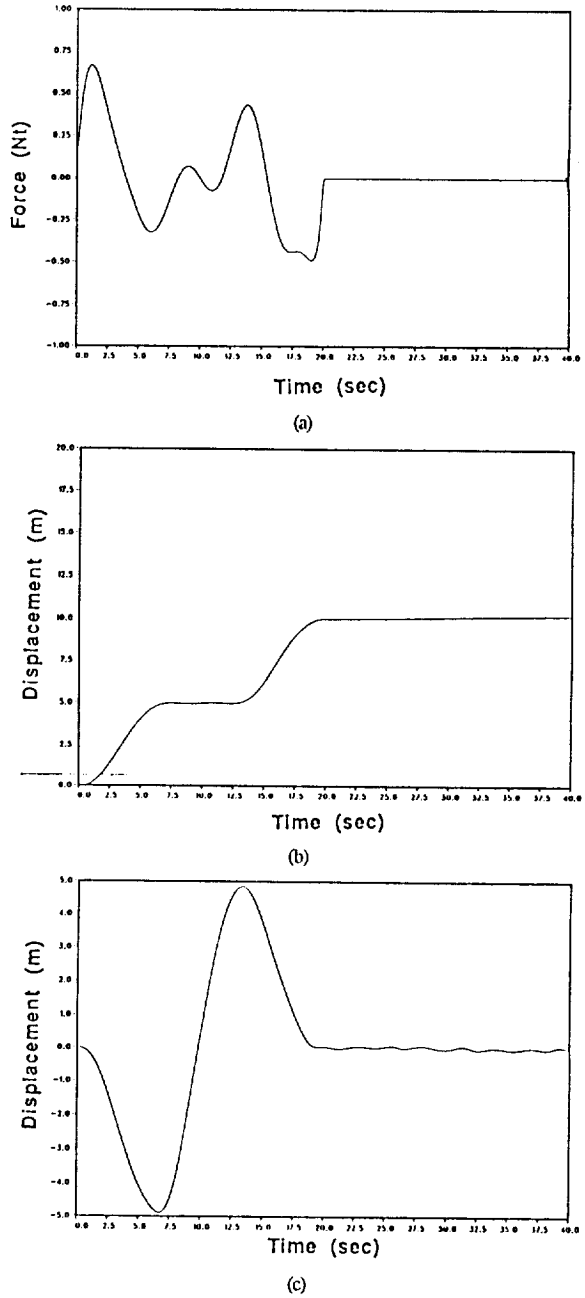


Fig. 6: Results of optimal maneuver of mass/flexible appendage system with reduced axial stiffness (case #3): (a) control force applied to rigid mass, (b) position of rigid mass, (c) deformation of tip of flexible appendage with respect to position of rigid mass.

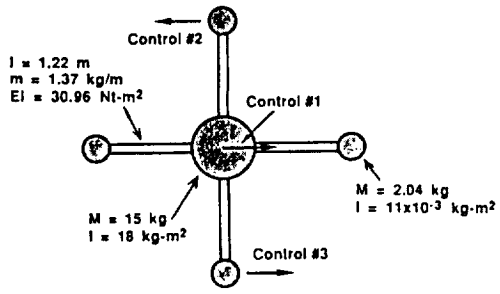


Fig. 7: Diagram of starfish configuration.

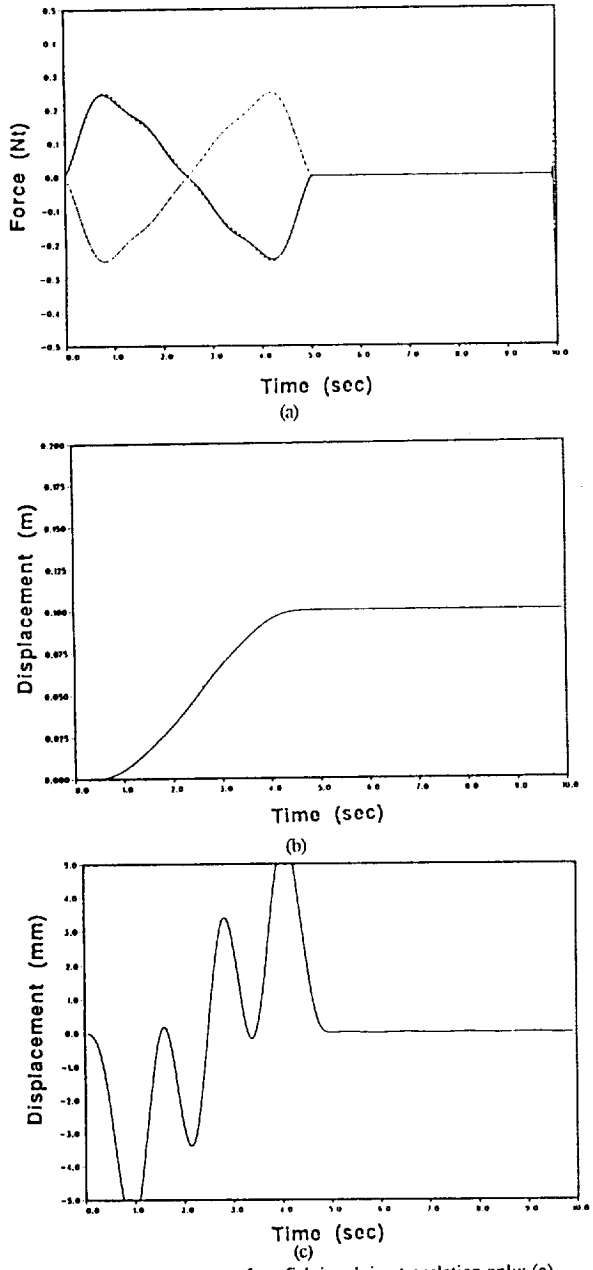


Fig. 8: Optimal maneuver of starfish involving translation only: (a) control forces applied at hub (solid line), thruster #2 (dotted line), and thruster #3 (double dotted line), (b) position of hub, (c) relative transverse deflection of flexible arm associated with control #2.

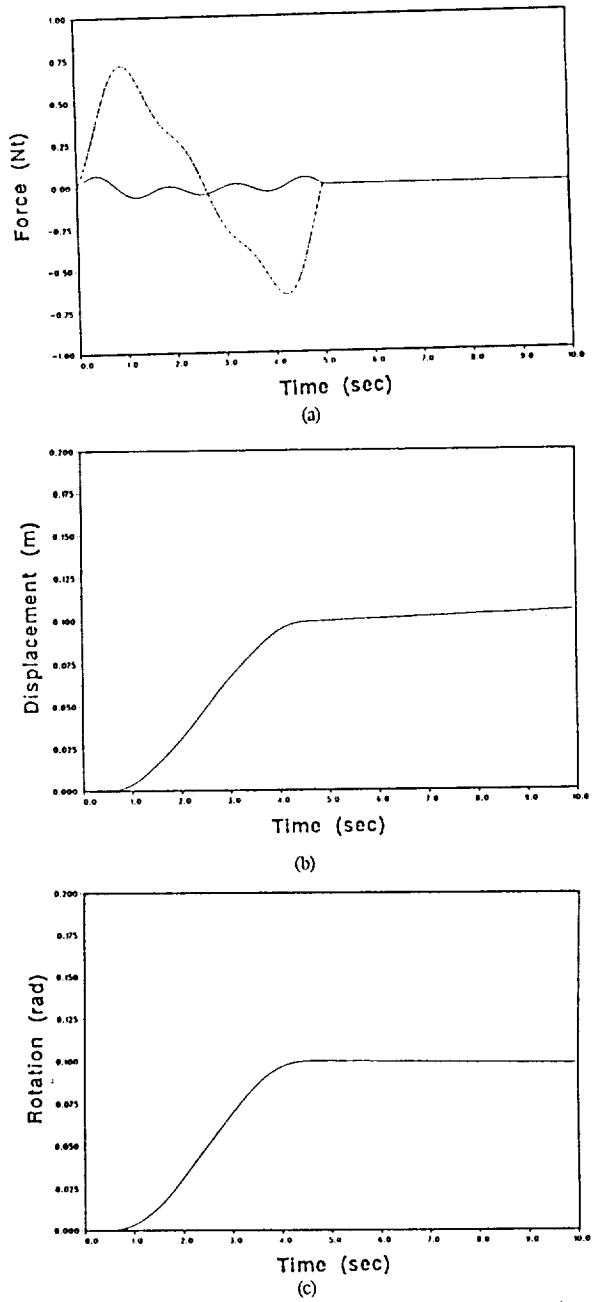
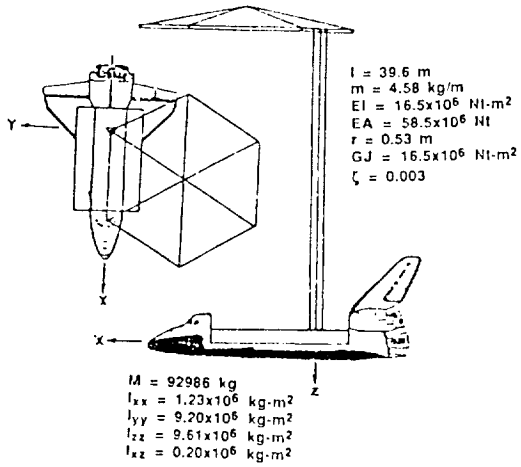


Fig. 9: Optimal maneuver of starfish involving translation and rotation: (a) Control forces applied to thruster #2 (solid line) and thruster #3 (dotted line), (b) translation of central hub, (c) rotation of central hub.

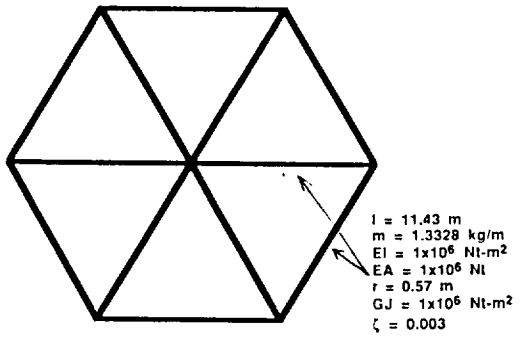
4.3 SCOLE Structure

The final example is a complex three dimensional structure proposed by NASA as a design challenge.[13] The Spacecraft COntrol Laboratory Experiment, shown in Fig. 10, consists of a rigid shuttle and hexagonal truss antenna connected by a flexible mast. Previous authors have treated the antenna as being rigid. In this paper, however, the flexibility of the antenna is considered. Figure 11 shows the transfer functions from a torque applied to the shuttle along about the axis of the mast to various points along the mast and antenna for both the rigid and flexible antenna models. A validation of this structural model with a traditional finite element code is

currently underway. As indicated in the figure, the flexibility of the antenna has a considerable effect on the transfer functions at higher frequencies. It is therefore appropriate to include a dynamic model for the antenna when applying the optimal control algorithm.

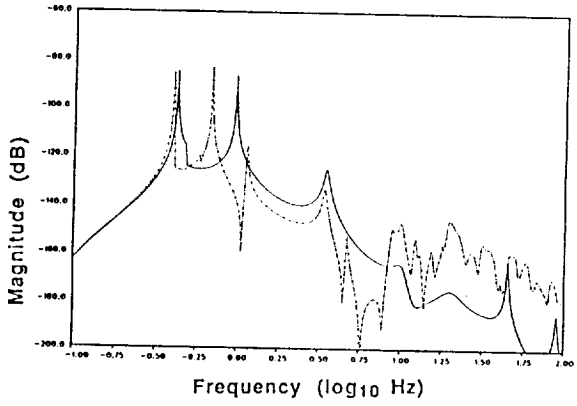


(a)

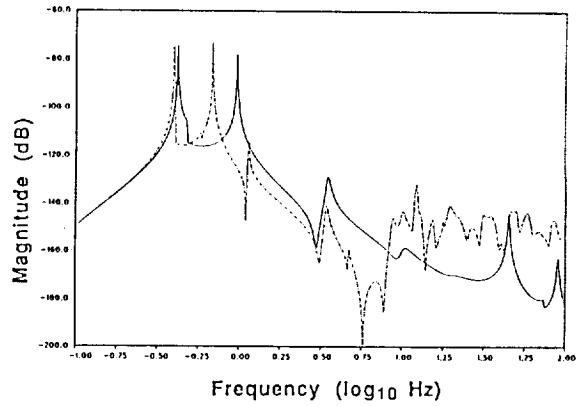


(b)

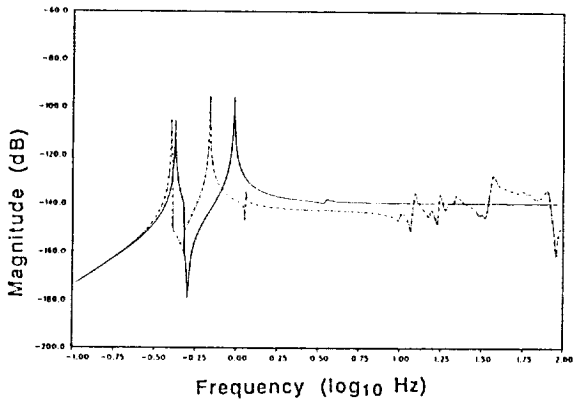
Fig. 10: The SCOLE experiment: (a) Shuttle and flexible mast properties, (b) flexible antenna model used in this study.



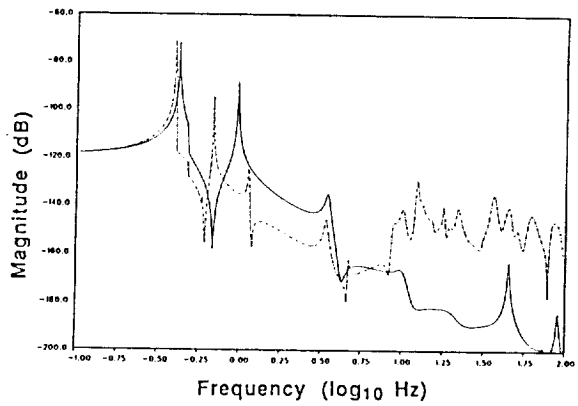
(a)



(b)



(c)



(d)

Fig. 11: Some transfer functions from control torque applied to shuttle along mast axis to inertial accelerations at various points on SCOLE structure for a rigid antenna (solid lines) and for a flexible antenna (dotted lines): (a) torsion of midpoint of mast, (b) torsion of mast at antenna junction, (c) transverse deflection of mast in pitching direction at antenna junction, (d) deflection, in plane of antenna, of antenna hub.

The SCOLE maneuver presented here consists of a 0.1 radian rotation about the z-axis of the shuttle. The three dimensional model used in the optimal control formulation incorporates 52 partial differential equations, which account for axial, torsional, and bending vibration in the mast and each of the twelve antenna elements. The shuttle is modeled as a massive rigid body with six degrees of freedom. Torque

controls directed along the z-axis are placed at either end of the mast. Due to the asymmetry of the structure, bending/torsion coupling is expected. Consequently, roll and pitch torque controls are also located on the shuttle. In the cost functional, equal magnitude and rate penalties are placed on all four controls. Large terminal penalties are applied to the roll, pitch and yaw angles of the shuttle, as well as the torsional deformation of the mast at its midpoint and at the mast/antenna junction. The maneuver time is ten seconds.

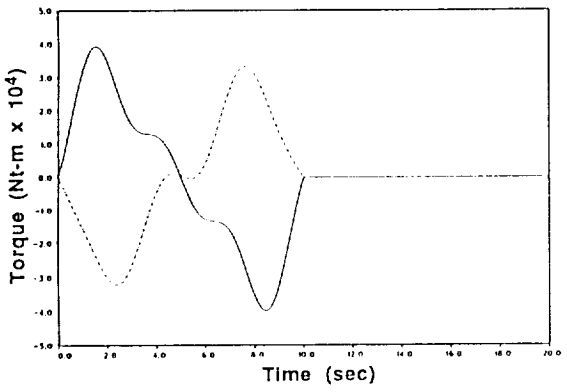
The results of the SCOLE slew are shown in Fig. 12. Because of the conventions used in defining the axes at the nodal points, the torques applied to the ends of the mast are opposite in sign. The physical torques are, however, applied in the same direction. From the figure, it is clear that, although the shuttle has rotated the prescribed amount, there is a small amount of residual torsional energy in the structure. This energy is due primarily to the deformation of the antenna. It is expected that additional penalties on structural deformation placed at various points on the antenna will significantly reduce this residual energy.

5. CONCLUSIONS

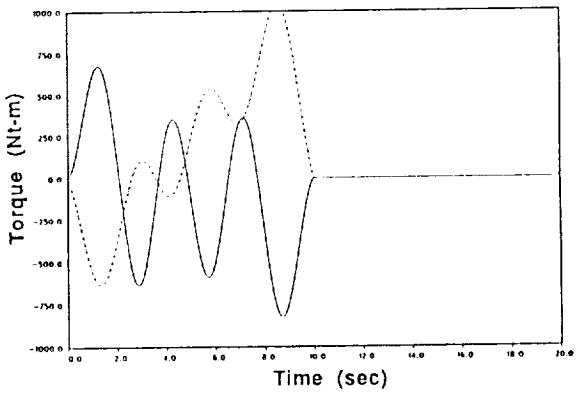
The open loop optimal control algorithm has been demonstrated for several structural models. The generality of the approach has been exploited in applying the algorithm to a complex structure. Furthermore, the ability of this approach to handle constraints and derivative penalties has been demonstrated. One major issue concerning this open loop method is modeling error. Because the approach presented is open loop, this formulation makes no guarantees on the performance of the actual structure, for which the mathematical abstraction is only an approximation. The necessary sensitivity analysis and closed loop formulations are topics of current research. It may also be possible to incorporate an adaptive identification algorithm to adjust the model in such a way as to minimize modeling error.

If the optimal control problem has an infinite time horizon, it is possible to use a different set of basis functions to converge on a solution. Candidate bases include Legendre and Laguerre functions, which are linear combinations of exponentials and other decaying functions.[14]

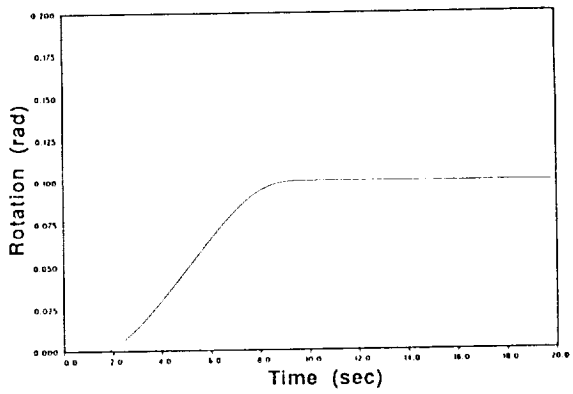
A method for incorporating distributed penalties into the optimal control algorithm is one topic of current research. Rather than penalizing certain points along a particular flexible element, an integral over the entire element, subject to some weighting function, becomes part of the cost functional. Such a capability would make it possible to develop controls that minimize the total (kinetic plus potential) energy within the structure in a continuous sense, rather than penalizing a large number of points within each flexible element.



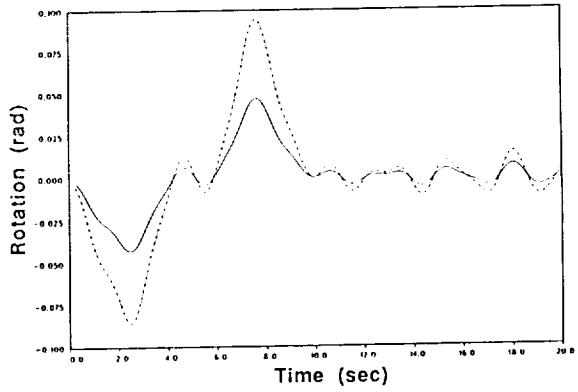
(a)



(b)



(c)



(d)

Fig. 12: Results of optimal SCOLE maneuver: (a) yaw torque applied to shuttle (solid line) and mast/antenna junction (dotted line), (b) roll torque (solid line) and pitch torque (dotted line) applied to shuttle, (c) yaw rotation of shuttle, (d) relative torsional deflection of mast at midpoint (solid line) and at mast/antenna junction (dotted line).

Work is currently underway to extend the basic modeling approach to continuous plate and membrane elements. This capability will permit modeling and control formulations for the NASA space station class of large space structures.

ACKNOWLEDGEMENTS

The work has been sponsored by the AFOSR, under contract number 49620-89-C-0082.

REFERENCES

[1] Ketner, G.L., "Survey of Historical Incidences with Controls-Structures Interaction and Recommended Technology Improvements Needed to Put Hardware in Space," Internal Document, Pacific Northwest Laboratory, PNL-6846, UC-222, March, 1989.

[2] Tzafestas, S.G. and Nightingale, J.M., "Optimal Distributed Parameter Control Using Classical Variation Theory," *International Journal of Control*, Vol. 12, No. 4, 1970, pp. 593-608.

[3] Miller, D.W., Hall, S.R., and von Flotow, A.H., "Optimal Control of Power Flow at Structural Junctions," *Proceedings of the 1989 American Controls Conference*, Pittsburgh, PA., June, 1989, pp. 212-220.

[4] MacMartin, D. and Hall, S.R., "An H_∞ Power Flow Approach to Control of Uncertain Structures," Space Systems Laboratory Report No. 10-89, Massachusetts Institute of Technology, Cambridge, Ma.; Sept., 1989.

[5] Breakwell, J.A., "Optimal Control of Distributed Systems," Paper No. 80-1737, presented at the *AIAA Guidance and Control Conference*, Danvers, Ma., August, 1980.

[6] Skaar, S.B., "Closed Form Optimal Control Solutions for Continuous Linear Elastic Systems," *Journal of the Astronautical Sciences*, Vol. 32, No. 4, Oct.-Dec. 1984, pp. 447-461.

[7] Pichè, R.A., "Analysis of Structural Control Problems Using Frequency-Domain Continuum Methods," PhD Thesis, University of Waterloo, Waterloo, Ontario, 1986.

[8] Wittrick, W.H. and Williams, F.W., "A General Algorithm for Computing Natural Frequencies of Elastic Structures," *Quarterly Journal of Mechanics and Applied Mathematics*, Vol. 24, No. 3, 1971, pp. 263-284.

[9] Hildebrand, F.B., *Advanced Calculus for Applications*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1976.

[10] Wilcox, D.J., "Numerical Laplace Transform and Inversion," *Int'l J. of Electrical Engineering Education*, Vol. 15, 1978, pp. 246-265.

[11] Kulla, P.H., "Continuous Elements - Some Practical Examples," *Proc. ESTEC Workshop, Modal Representation of Flexible Structures by Continuum Methods*, ESTEC, Noordwijk, The Netherlands, June 15-16, 1989, pp. 171-192.

[12] Bailey, T.L., "Distributed Parameter Control of a Cantilever Beam Using a Distributed Parameter Actuator," SM Thesis, Massachusetts Institute of Technology, Cambridge, Ma., Sept. 1984.

[13] Taylor, L.W., and Balakrishnon, A.V., "A Mathematical Problem and a Spacecraft Control Laboratory Experiment (SCOLE) Used to Evaluate Control Laws for Flexible Spacecraft .... NASA/IEEE Design Challenge," NASA Technical Memorandum 89075, Nov. 17-18, 1986, pp. 386-412.

[14] Hall, S.R., Private communication, April, 1990.

7538

N91-22326

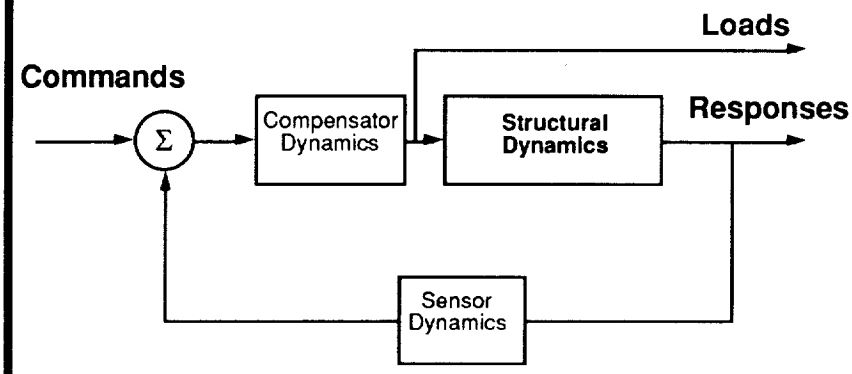
Structural Representation for Analysis of a Controlled Structure

Paul A. Blelloch
SDRC Engineering Services
San Diego, CA

Supported by the Space Station Program
NASA Lewis Research Center
Cleveland, Ohio

4th NASA Workshop on Computational
Control of Flexible Aerospace Systems
July 11-13, 1990

What Structural Representation Results in Accurate Closed-Loop Model?



The purpose of this study is to determine what reduced order structural representation is most appropriate for coupling with a control system. The goal is to choose a reduced order structural model which retains as closely as possible the characteristics of the closed-loop model with a full order structural representation. By characteristics of the closed-loop model we mean the closed-loop eigenvalues and the closed-loop transfer functions from commands to loads and from commands to responses. This process does not address the accuracy of the full-order model (usually a finite element model) but only the loss of accuracy associated with reducing the model. For the purposes of this study we will limit ourselves to collocated sensors and actuators. The choice of a structural representation for non-collocated sensors and actuators is not so clear.

What Do We Mean by Accurate Closed-Loop Model?

(1) Accurate Closed-Loop Frequencies:

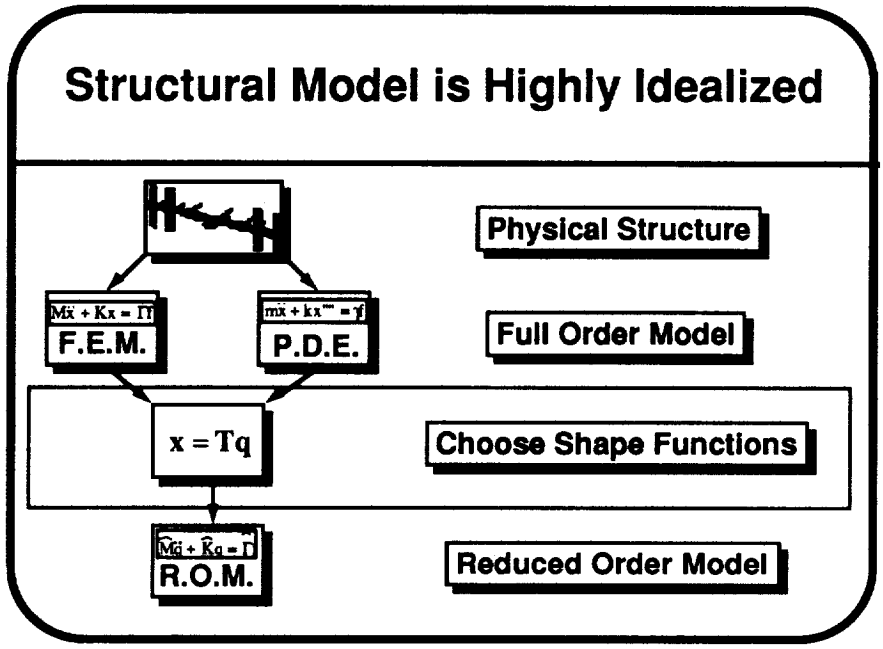
$$\varepsilon = \frac{|\lambda_{approx.} - \lambda_{exact}|}{|\lambda_{exact}|}$$

(2) Accurate Closed-Loop Frequency Response:

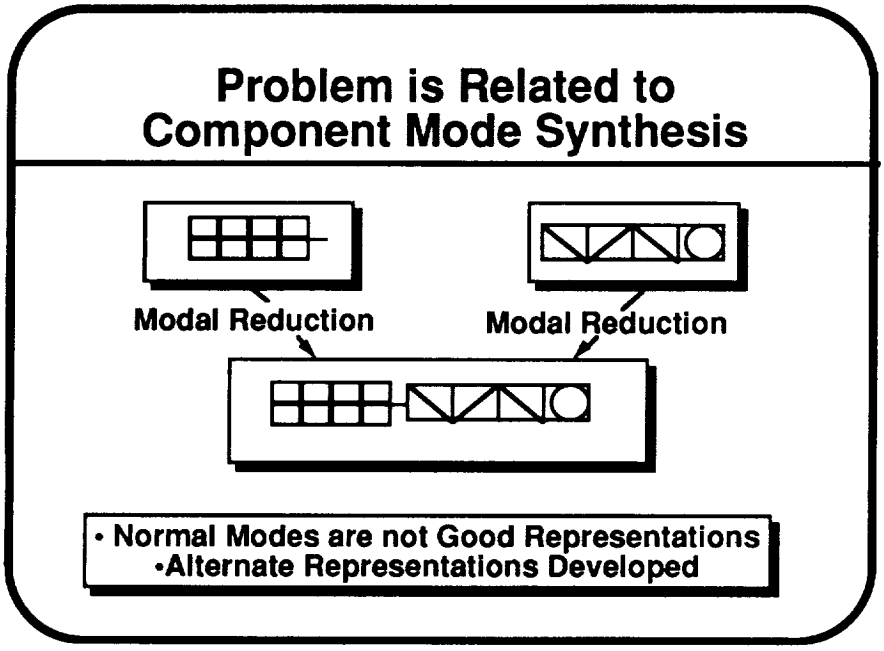
$$\varepsilon(j\omega) = \frac{|G(j\omega)_{approx.} - G(j\omega)_{exact}|}{|G(j\omega)_{exact}|}$$

- **From Commands to Responses**
- **From Commands to Input Loads**

More specifically we will define errors as follows: For each closed-loop frequency we will define a relative error as the distance from the closed-loop pole based on the reduced model to the closed-loop pole based on the full order model, divided by the magnitude of the closed-loop pole based on the full-order model. For the transfer functions from commands to input loads and from commands to responses we will use the same measure to define a relative error as a function of frequency. The reduced order model is said to closely represent the closed-loop model when these error measures are small.



As mentioned before we are only addressing one aspect of the accuracy issue. The overall modeling process starts with a physical structure or drawings of a physical structure and proceeds to develop what we will call a full-order model. Whether this model is based on finite elements or partial differential equations, a number of assumptions were made in its derivation. For the purposes of this study, we will assume that these assumptions are valid to the extent that the full-order model accurately represents modal frequencies in the control bandwidth and also the static deflection due static loads applied at the controller interface locations. The full order model, whether it is represented by a finite element model or partial differential equations, is almost certainly too large for practical control system analysis. The model is reduced by choosing a small number of shape functions (often normal modes). We are addressing the choice of these shape functions, such that accuracy of the closed-loop model is retained with as few functions as possible.



A very similar problem is that of component mode synthesis (CMS). In CMS the goal is to represent a number of substructures by reduced order models based on shape functions such that when these substructures are coupled, the modal frequencies of the coupled model will be as accurate as possible. In the case of control-structure interaction (CSI) we are simply replacing one the substructures by a control system.

Researchers in CMS have demonstrated for more than 20 years that the use of normal modes to represent the substructures can result in large inaccuracies of the coupled model. A number of alternate substructure representations have been developed that result in much more accurate coupled models.

Alternate Representations are Statically Exact

- **Residual Flexibility adds Static Contribution of Neglected Modes**
- **Craig-Bampton Representation adds Static Solution to Cantilevered Modes***
- **Lanczos Vectors are Based on Static Solutions**

***Implemented in NASTRAN as standard method**

Two methods for representing substructures in CMS have emerged as standards. These are normal modes with addition of residual flexibility and a Craig-Bampton representation. The use Lanczos vectors rather than modes has also been suggested.

The residual flexibility method adds static flexibility that is not represented by the retained normal modes. This flexibility can either be represented as a purely static flexibility at the interface or by a high frequency subsystem which contributes quasi-statically at the interface. The residual flexibility subsystem is uncoupled (orthogonal to) the retained normal modes.

The Craig-Bampton method combines a static reduction (Guyan reduction) to the interface degrees of freedom with a set of normal modes calculated with the interface degrees of freedom held fixed.

Lanczos vectors are generated by a series of static solutions and do not require the solution of an eigenvalue problem. The resulting mass and stiffness matrices are tri-diagonal rather than diagonal.

Craig-Bampton Representation Standard in Structural Dynamics

$$\begin{bmatrix} M_{BB} & M_{IB} \\ M_{IB}^T & I \end{bmatrix} \begin{Bmatrix} \ddot{x}_B \\ \ddot{x}_I \end{Bmatrix} + \begin{bmatrix} K_{BB} & 0 \\ 0 & \Omega^2 \end{bmatrix} \begin{Bmatrix} x_B \\ x_I \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$$

- **Static shapes based on unit deflections of exterior DOF**
- **Guyan reduction to exterior DOF**
- **Static shapes = rigid body modes for rigid body control**
- **Relative not absolute DOF fixed at joints**
- **Results in accurate system models**

The Craig-Bampton method is the most popular method used in CMS. It is conceptually very simple, it is accurate and it is implemented as the standard representation used by MSC/NASTRAN's superelement capability. Two sets of shape vectors are used. The first are static shapes based on unit deflections at the interface. These are the same shape functions used in the Guyan reduction process. The second are normal modes calculated with the interface fixed. The Craig-Bampton representation has the form illustrated above. While it is not diagonal, it can be diagonalized. In this case it is very similar to the residual flexibility representation, with a number of normal modes combined with a set of high frequency modes representing static flexibility at the interface.

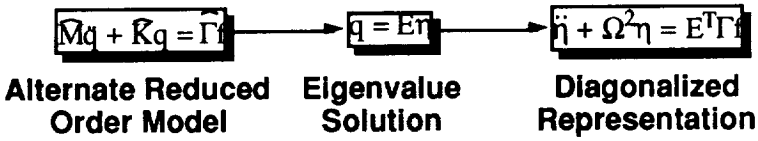
When the Craig-Bampton representation is used in CSI for systems with joints, the relative rotation rather than the absolute rotation at the joint can be held fixed during the calculation of "component" modes.

Why Do We Persist in Using Normal Modes?

- Tradition
- Obtainable from any Structural Dynamics Routine
- Physical Interpretation
- Approximately Balanced (in sense of Moore)
- We Don't Understand Damping (modal damping)
- Small Amount of Data to Transfer
- Uncoupled Equations of Motion

Given that normal modes are known to generate poor solutions in CMS problems, and given that alternate representations are better, why are normal modes used so pervasively in CSI? There are a number of reasons why normal modes are convenient. They are standard output from any structural dynamics routine, and are certainly more "standard" than the alternate representations used in CMS. They have a physical interpretation. For lightly damped systems with sufficiently separated frequencies they are approximately balanced and cost decoupled, suggesting that they are natural coordinates to use for model reduction. Modal damping is simple to define. The amount of data transferred is limited to modal frequencies and mode shape coefficients. And finally, the resulting equations of motion are uncoupled, resulting in very fast simulations.

Alternate Representations can be Diagonalized



Alternate Reduced Order Model | **Eigenvalue Solution** | **Diagonalized Representation**

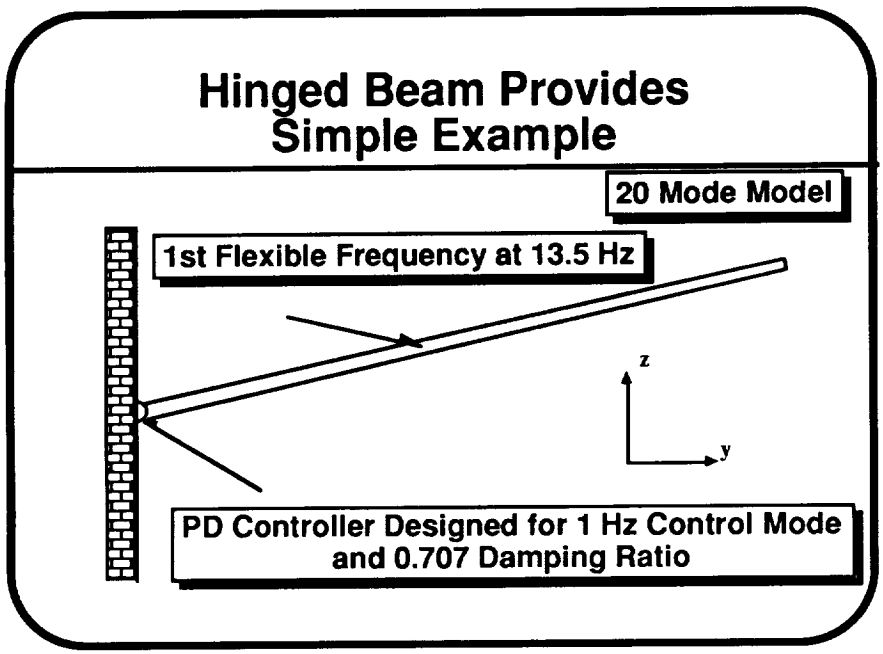
- **Eigenvalue Problem is Small (size of reduced order model)**
- **Diagonalized Representation has Many Advantages of Normal Mode Representation**

Many of the advantages of the normal modes representation are shared by the Craig-Bampton representation, if the reduced order model is diagonalized. The diagonalization involves an eigenvalue solution on the reduced order model, which is typically very fast. The resulting "modes" will include some low frequency normal modes along with some high frequency residual modes which contribute quasi-statically in the low frequency range. The coordinates are now balanced and cost-decoupled with respect to the reduced order model, though not necessarily with respect to the full order model. The amount of information to be transferred is again limited to frequencies and mode shape coefficients and the equations of motion are uncoupled. The high frequency modes may need to be treated carefully during a transient simulation, though extra damping can be added without affecting their contribution in the frequency range of interest.

Summary of Structural Representations

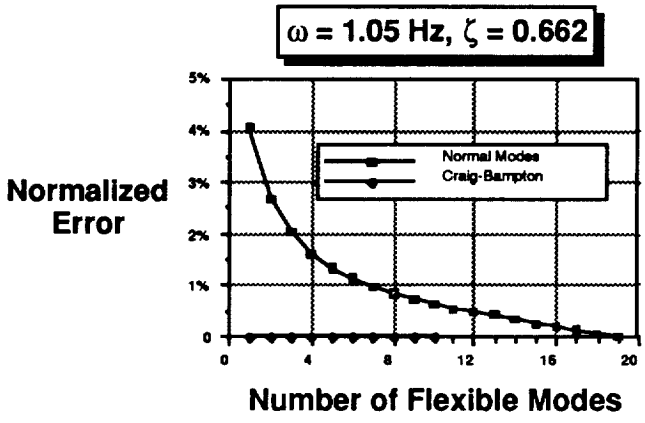
- Goal is Accurate Closed-Loop Model
  - Accurate Closed-Loop Frequencies
  - Accurate Closed-Loop Transfer Functions
- Choice of Shape Functions can be Motivated by CMS
- Alternate Representations are Statically Exact
- Alternate Representations can be Diagonalized

In summary, the goal of this study is to select a minimal number of shape functions that accurately represent the closed-loop model. While normal modes are often used, alternate representations developed in the field of CMS can also be applied. These alternate representations are statically exact at the interface points and can be diagonalized to recover some of the advantages of normal modes. Following, we will show two examples which demonstrate that the Craig-Bampton representation does in fact result in significantly more accurate closed-loop models than the normal modes representation.



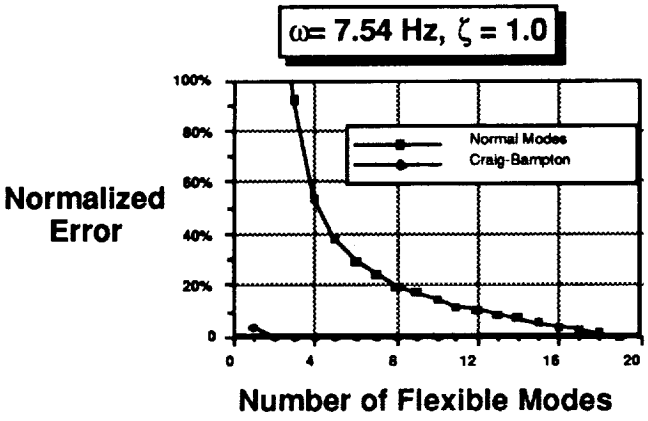
One might expect that as a control system became stiffer with respect to the structure, a set of fixed interface modes would be more appropriate, while for a soft control system the free interface modes would be more appropriate. It is certainly true, that as the control system gets stiffer, the inaccuracies associated with the use of free interface modes become larger, however, this simple example shows that the errors can be large even when the control system is significantly softer than the structure. We have chosen PD control gains to give a rigid body frequency of 1 Hz and a damping ratio of 70.7%. The actual frequency and damping ratio will differ due to the flexibility of the beam. The "full-order" model is the finite element model with 20 degrees of freedom.

Craig-Bampton Modes Result in More Accurate Control Frequency



The actual control frequency based on the full 20 degree of freedom model is 1.05 Hz with a damping ratio of 66.2%. Using even one Craig-Bampton mode results in an exact representation of the frequency, while seven normal modes are required to reduce the error to less than 1%.

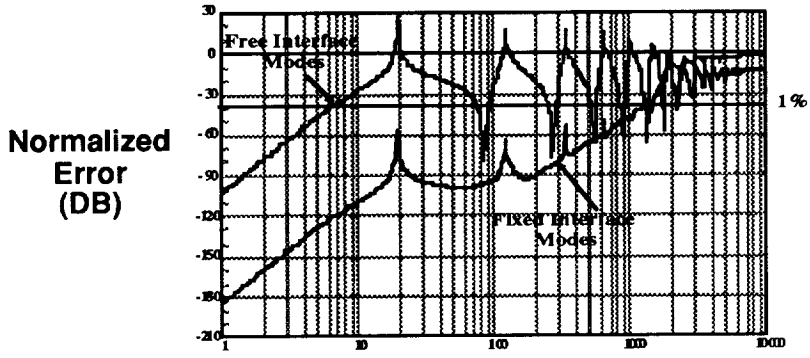
Craig-Bampton Modes Result in More Accurate 1st Flexible Frequency



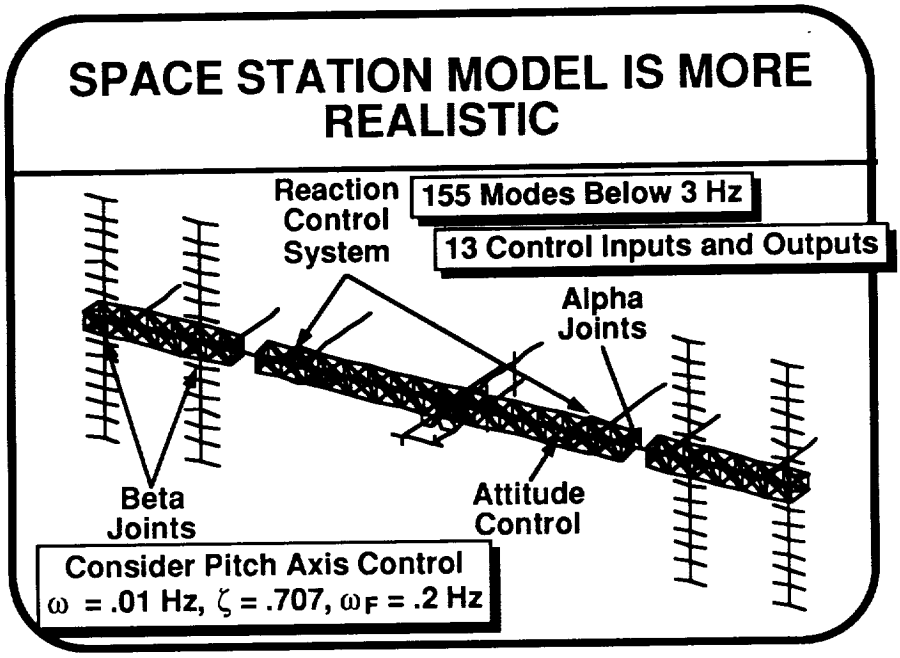
The first closed-loop flexible mode is at 7.54 Hz and is critically damped. In this case two Craig-Bampton modes represent the closed-loop mode exactly, while thirteen normal modes are required to reduce the error to less than 10%. In this case the error in the closed-loop frequency using normal modes is drastic.

Craig-Bampton Modes Result in More Accurate Transfer Function

Closed-Loop Transfer Function from a Rotational Command to a Rotational Response



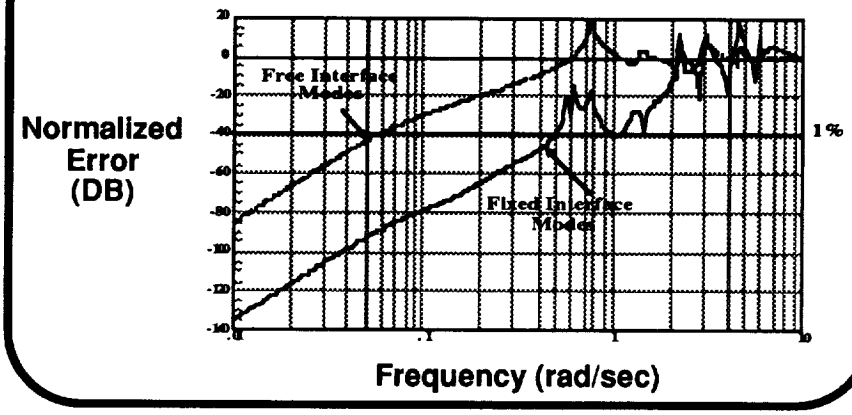
Examining the frequency response from a rotational command to a rotational response tells the same story. Using Craig-Bampton modes, the error is less than 1% up to a frequency of 1000 rad/sec, while using normal modes, the error exceeds 1% at just over 6 rad/sec.



The methods presented here were developed for Space Station Freedom, which is a complex structure with very high modal density. Examining pitch control of the Space Station provides a more realistic example. In this case the first significant flexible mode interacting with the control system is near 0.2 Hz. The control gains are chosen to provide a rigid body control frequency of 0.01 Hz and a damping ratio of 70.7%. Once more, this is a system where the control frequency is more than an order of magnitude below the first flexible frequency. In this case it is not possible to calculate the true "full-order" closed-loop model because the finite element model has over 1000 degrees of freedom, so the exact model is one based on 155 Craig-Bampton modes. The two reduced order models are each based on 41 flexible modes. The first based on 41 Craig-Bampton modes, and the second based on 41 normal modes.

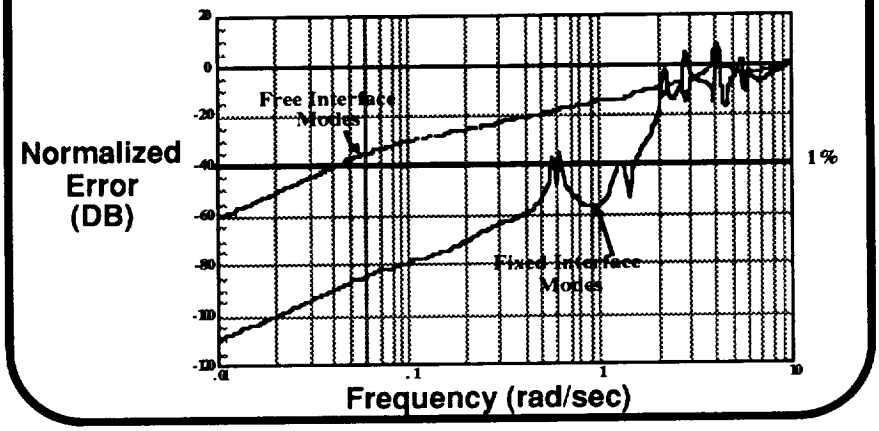
Craig-Bampton Modes Result in More Accurate T.F. to Response

Transfer Function from Pitch Command to Pitch Response



The error in the transfer function from a rotational command to a rotation about the pitch axis again illustrates the improved accuracy associated with the Craig-Bampton representation. In this case the reduced order model based on 41 Craig-Bampton modes is accurate (less than 1% error) up to a frequency of 0.5 rad/sec, while the reduced order model based on 41 normal modes is only accurate up to a frequency of 0.05 rad/sec.

Craig-Bampton Modes Result in More Accurate T.F. to Torque



Results for the transfer function from rotational commands to torques applied by the controller to the structure suggest similar conclusions. Again the model based on Craig-Bampton modes is significantly more accurate than the model based on normal modes.

CONCLUSIONS

- **Alternate Modal Representations are Available and Easy to Implement**
- **Key Difference is Exact Static Representation**
- **Alternate Modal Representations can be Diagonalized**
- **Alternate Modal Representations Result in More Accurate Closed-Loop Models with Collocated Sensors and Actuators**
- **Non-Collated Sensors and Actuators Less Clear**

In conclusion, the use of an alternate structural representation, such as the Craig-Bampton representation, can result in much more accurate results than are obtained when using a truncated set of normal modes. The key difference between the alternate representations and the normal mode representation is the incorporation of a static solution. The alternate representations do not necessarily generate diagonal mass and stiffness matrices, but they can be diagonalized at a minimal effort in order to capture some of the advantages of normal modes.

All the results presented here are based on collocated sensors and actuators. The issue with non-collocated sensors and actuators is somewhat different since it is not clear which points should be held fixed during the modal solution. This is an issue that still needs to be resolved.

PDEMOD-Software for Controls-Structures Optimization

Lawrence W. Taylor, Jr.
and
David Zimmerman

4th NASA Workshop on Computational Control of Flexible Aerospace Systems

PDEMOD - Software for Control/Structures Optimization

By
Lawrence W. Taylor, Jr.
NASA Langley Research Center
and
David Zimmerman
University of Florida

ABSTRACT

Because of the possibility of adverse interaction between the control system and the structural dynamics of large, flexible spacecraft, great care must be exercised to ensure stability and system performance. Because of the high cost of insertion of mass into low earth orbit, it is prudent to optimize the roles of structure and control systems simultaneously. Because of the difficulty and the computational burden in modeling and analyzing the control/structure system dynamics, the total problem is often split and treated iteratively. The awkwardness and inaccuracy of this approach can lead to poor designs.

It would aid design if the control/structure system dynamics could be represented in a single system of equations. Heretofore this has not been possible. With the availability of the software PDEMOD it is now possible to optimize structure and control systems simultaneously. The distributed parameter modeling approach enables embedding the control system dynamics into the same equations for the structural dynamics model. By doing this the current difficulties involved in model order reduction are avoided.

The NASA Mini-MAST truss is used as an example for studying integrated control/structure design. Comparisons are made for (1) structure without control, (2) controls only, and for optimal combinations of structure and control. Both proof-mass and torque-wheel actuators are considered. The results give insight with regard to the essential factors in trading structure and control for space applications.

INTRODUCTION

Certain future space missions will be performed by large, flexible spacecraft. Because of the high cost of insertion of mass into low earth orbit, it is prudent to optimize the design of both the structure and control systems. The current practice is to create a finite element model of the structure and to use a reduced order modal model for control synthesis. Unfortunately, this disjoint procedure is inaccurate and is an impediment to integrated design. There is a need for a dynamics model

which includes both structural and control dynamics which is parameterized in terms of the design variables.

Distributed parameter modeling has been shown to be quite accurate for modelling (reference 1.) the dynamics of the first six modes of portal frames. The root-mean-square error in frequency was about 7/10 of one percent for the first six modes of two experimental configurations. The difficulty in modeling complex structures has been an obstacle to the use of distributed parameter modeling. The development of finite element software has resulted in the wide-spread practice of modeling flexible structures with finite element models. The availability of modeling software such as DISTEL (reference 2.) and BUNVIS-RG (reference 3.) offers the alternative of modeling complex configurations with distributed parameter models.

Another advantage of distributed parameter modeling is that control and sensor dynamics can be incorporated into the equations of motion of the structural dynamics. Again, the burden of assembling the necessary equations for complex configurations has become routine because of the capabilities of the software PDEMOD (reference 4). It is now possible to optimize structure and control systems simultaneously (references 5. and 6.) for complex spacecraft because of the incorporation of their dynamics into a single system of equations. The distributed parameter modeling embeds the control system dynamics into the same equations for the structural dynamics model so that model order reduction is not necessary. The resulting structures/controls model is particularly well suited for integrated design.

In this paper only preliminary comparisons are made for the Mini-MAST truss (references 7. and 8.) for (1) structure without control, (2) controls only, and for optimal combinations of structure and control. Both proof-mass and torque-wheel actuators are considered. The results give insight into the essential factors in trading structure and control for space applications. Subsequent, more in-depth study will consider the transient dynamics aspects using PDEMOD and will use the optimization techniques of Fogel and Holland (references 9. and 10.).

DISCUSSION

The software PDEMOD enables the generation of models of complex structural configuration which include the dynamics of the control system as well as the structural dynamics. This is done using partial differential equations to describe the dynamics of flexible beam elements which together with rigid body elements form a connected network of components of the structure. The coefficients of the sinusoidal and hyperbolic functions for each flexible element give the mode shapes. The sensed motion and control forces and moment are expressed in terms of

the same parameters and the influence of control on the configuration is represented by terms added to the equations for the structural dynamics.

Structural Dynamics

First, the structural configuration is viewed as an assembly of flexible and rigid elements. It is then necessary to indicate the connectivity of the elements. This is done by giving the identification of the rigid bodies at either end of each flexible beam element and the related points of attachment. The alignments of the flexible beam axes must also be given. The axis for each rigid body is that of a particular, attached beam.

The next input needed is the stiffness (EI_x, EI_y, EA_z, EI_y) and mass (m/L, I_y/L) characteristics and the length of each flexible beam element. The mass and inertia of each rigid body is needed to complete the information required for the structural dynamics model.

Control Dynamics

The addition of feedback control does not increase the order of the system matrix unless the applied force or moment is applied to the interior of a flexible beam element. In such a case it is necessary to add a node or rigid body with negligible mass at that point. In other cases it is only necessary to add terms to the matrix elements which already exist. The dynamics of the sensors and actuators are inserted as transfer functions multiplying the additional terms. The additional terms are located in the rows which correspond to the body to which the control force and/or moment is applied, and in the columns which correspond to the beam elements which contain the location of the sensed motion.

Optimization

In order to perform optimization for parameter estimation or for criteria involving structural dynamics, sensitivity functions are usually required. The sensitivity functions relate the change in the criterion to changes in the parameters involved in the optimization. Although it is possible to express in closed form these derivatives it is most unwieldy to evaluate the expressions. This is because derivatives must be taken of the determinant of the system matrix which can be quite large. It is more practical to approximate the derivative numerically by the ratio of the change in the criterion to the change in the parameter. This was the approach used in reference 8. for a parameter estimation application. Changes in the modal frequencies for each of the model parameters was generated numerically.

Although parameter estimation is an example of optimization, it is the optimal, integrated control/structural design that we now turn our attention. The selection of the design criterion and the corresponding conditions or constraints are of paramount importance. An ill-posed problem can easily result in nonsensical results which bear no relation to the actual design problem. Perhaps the most suitable design criterion for the integrated control/structure problem would be the life-cycle-cost of the entire system, subject to a set of system performance and structural specifications. In many cases it suffices to consider only the total structure and control system mass.

There are alternatives to optimization schemes which require the sensitivity functions mentioned earlier. Genetic algorithms (GA's), as introduced by Holland (reference 10), are one form of directed random search. The form of direction is based on Darwin's theory of the "survival of the fittest". In GA's a finite number of candidate solutions or designs are randomly (or heuristically) generated to create an initial population. This initial population is then allowed to evolve over generations to produce new, and hopefully better designs. The basic conjecture behind GA's is that evolution is the best compromise between determinism and pure chance. GA's have the capability to solve continuous, discrete, and mixed optimization problems.

There are four main operations in a basic GA: evaluation, selection, crossover, and mutation. Evaluation is the process of assigning a fitness measure to each member of the current population. The fitness measure is typically chosen to be related to the objective function which is to be maximized. No gradient or auxiliary information is used. Therefore, GA's are more likely to converge to a global maximum than a hill climbing algorithm, although no algorithm can guarantee convergence to the global maximum. Selection is the operation of choosing members of the current population to be parents for producing the next generation. Selection is weighted towards the more fit members of the population. Therefore, designs which are better as viewed from the fitness function, and therefore the objective function, are more likely to be chosen as parents. Crossover is the process in which design information is transferred to the prodigy from the parents. Mutation is a low probability random operation which slightly perturbs the design represented by the prodigy. The mutation operation is used to retain design information over the entire domain of the design space during the evolutionary process.

INTEGRATED DESIGN PROBLEM

The integrated control/structural design problem to be considered is to minimize the total system mass while limiting the response to a disturbance force at the tip of the Mini-MAST truss. This will be accomplished by the selection of the stiffness of the truss elements, the

use of a proof mass actuator and a torque wheel actuator, both located at the tip. The total system mass is increased when (1) the stiffness of truss elements is increased, (2) the capacity of the proof mass actuator is increased, and (3) the capacity of the torque wheel actuator is increased. Prior to involving the software PDEMOD to calculate the dynamic response of the actively controlled Mini-MAST to the disturbance force, it is prudent to investigate the best mix of structural stiffness and the use of active control in a more simple way.

The structural stiffness of a uniform Bernoulli beam is given by:

$$K = 3EI/L^3$$

Because the mass of the truss elements is proportional to their stiffness the truss mass is:

$$Mass_{truss} = Mass_{truss,o}[EI/EI_o][L_o/L]^3$$

Note that to keep the same resistance to a disturbance force applied at the tip of the Mini-MAST the increased stiffness and corresponding mass increases by the length to the third power.

Because the 66-foot Mini-MAST truss has 18 bays it is possible to alter the stiffness of the truss elements as each bay to reduce the total mass while achieving the same stiffness or resistance to a disturbance force at the tip. The maximum saving that can be achieved appears to be about 17 percent.

A moving or proof mass actuator (PMA) can produce a force to oppose a constant disturbance force but only for a relatively short period of time which depends on the size of the proof mass and its stroke. Neglecting the stationary mass the mass for the PMA can be shown to be:

$$Mass_{PMA} = .5*Force*Time^2/Stroke$$

Note that the mass of the proof mass actuator is proportional to the time of application squared but does not depend on the truss length or stiffness.

A rotating mass or torque wheel actuator (TWA) can produce a moment which by reaction of the truss structure can oppose a constant disturbance force but only for a relatively short period. The mass of the TWA depends on the moment, maximum wheel speed and the time of application. Neglecting the stationary mass the mass of the TWA can be expressed as:

$$Mass_{TWA} = Moment*Time/(Wheel\ Speed*Radius^2)$$

A moment applied to the tip of the Mini-MAST produces a lateral deflection which can be used to oppose a disturbance force. The moment per deflection is given by:

$$\text{Moment/Deflection} = 2EI/L^2$$

The force per deflection is:

$$\text{Force/Deflection} = 3EI/L^3$$

The mass of the totque wheel actuator (TWA) capable of countering a particular force for a specific time becomes:

$$\text{Mass}_{TWA} = .667*\text{Force}*\text{Time}*\text{Length}/(\text{Max Wheel Speed}*\text{Radius}^2)$$

Note that the mass of the torque wheel actuator is proportional to both the time of application of the disturbance force and the length of the Mini-MAST.

The problem now is to determine the combination of structural stiffness and the sizes of the proof mass and torque wheel actuators which minimizes the total mass.

$$\text{Mass}_{Total} = \text{Mass}_{truss} + \text{Mass}_{PMA} + \text{Mass}_{TWA}$$

The fundamental variables which determine the best mix is the time, T, for which the disturbance force is applied and the length, L, of the Mini-MAST truss. By examining different combinations of truss length, L and the time, T, for which the disturbance force is applied the regions for which the truss structure, the proof mass actuator, or the torque wheel actuator is best in countering the disturbance force can be determined. The result of such a study shows that the truss structure is best for large values of the time for which the disturbance force is applied. The proof mass actuator is best for long trusses. Under only limited conditions is the torque wheel actuator best for countering disturbance forces.

CONCLUDING REMARKS

Distributed parameter models of structures have important advantages for problems involving the active control of flexible structures. This is especially true for repetitive lattice structures such as the Mini-MAST truss because its dynamics can be accurately represented by only a few parameters.

Until the advent of software for distributed parameter modeling of complex flexible structures it was not practical to model complex spacecraft configurations using distributed parameter models. Now the software PDEMOD enables modeling complex configurations and can also

include the control system dynamics in the same equations. The need to resort to model order reduction is eliminated because the closed loop stability and system performance can be determined without ignoring any of the modes.

This capability enables working integrated control/structures problems. An example problem is examined in which structure and active controls are used to counter the disturbance force at the tip of the NASA Mini-MAST truss. The regions are determined in which structure, proof mass actuators, and torque wheel actuators result in the minimum mass system. Variations of this tradeoff between control and structure considerations are being pursued.

The development of the PDEMOD software has been underway for about one year. The formulation and coding has been completed for modelling general three-dimensional configurations. Modal frequencies and mode shapes have been generated for the Spacecraft Control Laboratory Experiment (SCOLE) configuration and the Mini-MAST truss. Graphics for drawing wireframes of the deflected configurations has been the most recent addition. Transient response, transfer functions, Timoshenko beam option and improved root-finding algorithms will be added during the next year. Copies of the source code will be made available to anyone interested in modelling new configurations or contributing to the software development.

REFERENCES

1. Bishop, Richard Evelyn Donohue and Daniel Cowan Johnson. (1962). Mechanics of Vibration. University Press. Cambridge, England.

2. Poeleart, Daniel. (1983). DISTEL, a Distributed Element Program for Dynamic Modelling and Response Analysis of Flexible Structures. Proceedings of the 4th VPI&SU/AIAA Symposium. Blacksburg, Virginia.

3. Anderson, Melvin S. and F. W. Williams. (1987). BUNVIS - RG: Exact Frame Buckling and Vibration Program, with Repetitive Geometry and Substructuring. Journal of Spacecraft and Rockets, Vol. 24, No. 4, July-August 1987, page 353.

4. Taylor, Lawrence W., Jr. (1989). Distributed Parameter Modelling for the Control of Flexible Spacecraft. The Proceedings of the NASA-UCLA Workshop on Computational Techniques in Identification and Control of Flexible Flight Structures. Lake Arrowhead, California.

5. M. Salama, J. Garha, L. Demsetz and F. Udwadia. (1988). Simultaneous Optimization of Controlled Structures. Computational Mechanics, Vol. 3, pp. 275-282.

6. Balakrishnan, A. V. (1990). Combined Structures-Controls-Integrated Optimization Using Distributed Parameter Models. Proceedings of the Workshop on Computational Techniques in Identification and Control of Flexible Flight Structures, Lake Arrowhead.

7. Pappa, Richard; Bob Miserentino, Jim Bailey, Ken Elliott, Sharon Perez, Paul Cooper and Boyd Williams. (1990). Mini-MAST Control Structure Interaction Testbed: A User's Guide. NASA TM -102630.

8. Taylor, Lawrence W., Jr. (1990). PDEMOD - Computer Software for Partial Differential Equation Modeling of Flexible Spacecraft. Proceedings of the 2nd USAF/NASA Workshop on System Identification and Health Monitoring of Precision Space Structures. Pasadena, California.

9. Fogel, Lawrence J., Alvin J. Owens and Michael J. Walsh. (1966). Artificial Intelligence Through Simulated Evolution. New York, Wiley.

10. Holland, J. H. (1975). Adaption in natural and artificial systems. University of Michigan Press, Edition No. 1, 1975. Ann Arbor, MI.

11. Taylor, Lawrence W., Jr. and Robert S. Dunning. (1985). Using a Life-Cycle-Cost Criterion for Multi-Disciplinary Design Studies for the Manned Space Station. AIAA-85-0221. AIAA 23rd Aerospace Sciences Meeting. Reno, Nevada.

INTRODUCTION

- Difficulties of Current Practice
- Advantages of Distributed Parameter Modeling
- Difficulties of Modeling Complex Structures
- Capabilities of PDEMOD Software
- Integrated Design Objective Functions
- Necessary Additions to PDEMOD
- Example Controls-Structures Design Problem
- Insights Offerred by Particular Tradeoffs
- Concluding Remarks

Issues in Modeling Complex Structures

Finite Element Modeling

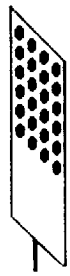
- Excessive Complexity
- Parameter Estimation is Difficult
- Model Order Reduction Required for Control Analysis

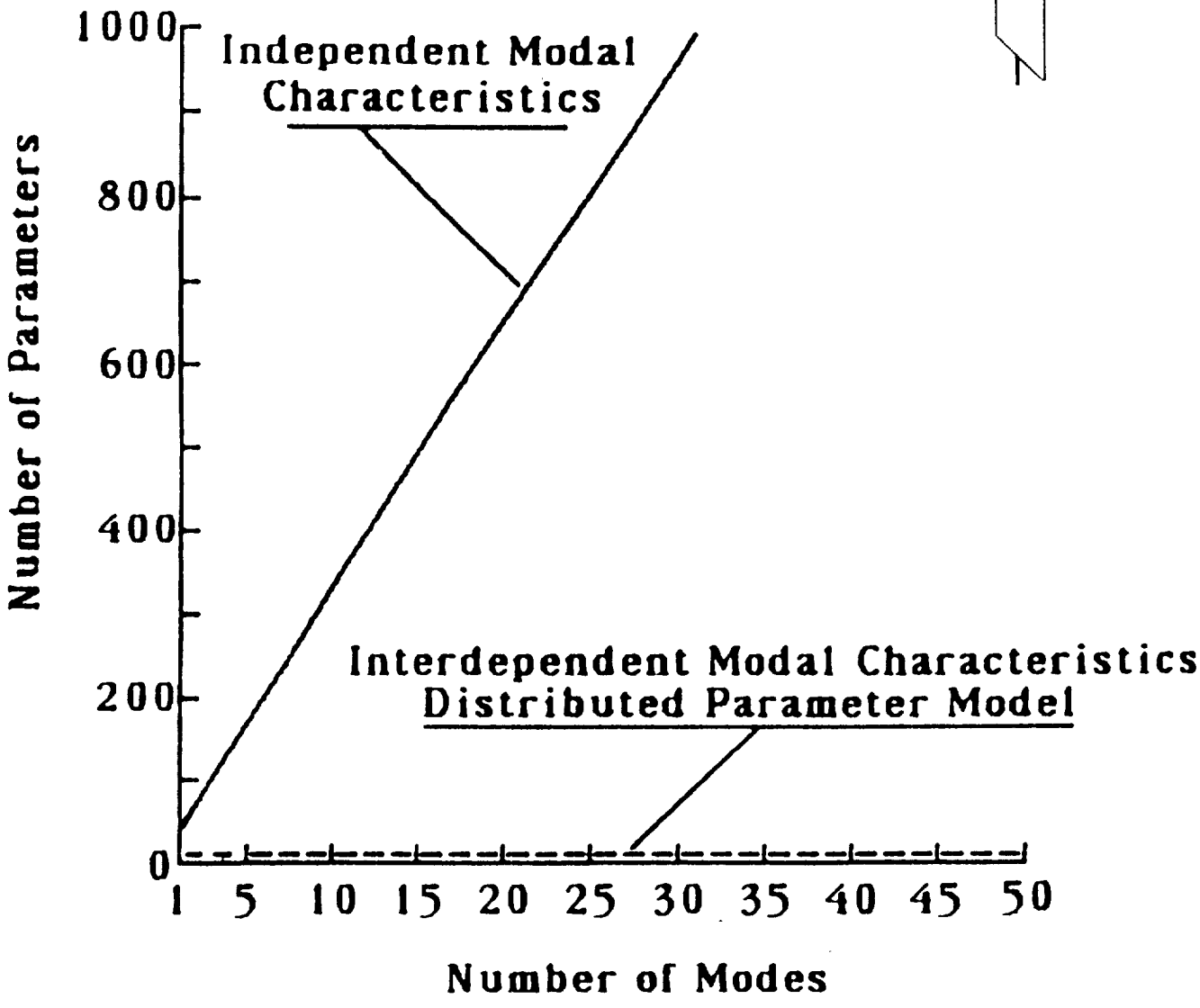
Distributed Parameter Modeling

- Fewer Model Parameters
- Parameter Estimation Straightforward
- Closed-Loop Stability Analysis does not Require Order Reduction

Number of Model Parameters

Solar Array Example:





Modeling Software

Name	Structural Dynamics	Optimization*
DISTEL *by Poelaert*	✓	
BUNVIS *by Anderson*	✓	
PDEMOD *by Taylor*	✓	✓

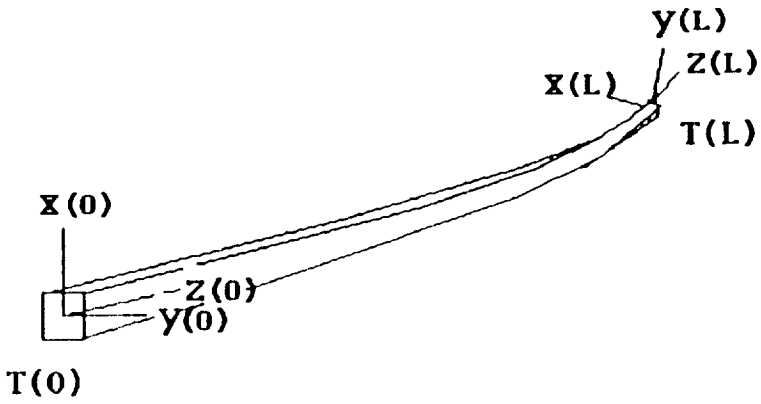
* — Parameter Estimation,
Control-Synthesis, and
Structural Design

New Modeling Software

PDEMOD

- Distributed Parameter Modeling
- Flex Beam Elements/Rigid Bodies
  (Bend-x, Bend-y, Twist, Stretch)
- Parameter Estimation
- Structural Damping
- Control System Dynamics
- Parametric Studies

Beam Model



The Moments and Forces at (0) in Beam Axes are:

$$M_x = EI_y u_y''(0) \qquad F_x = EI_y u_y'''(0)$$

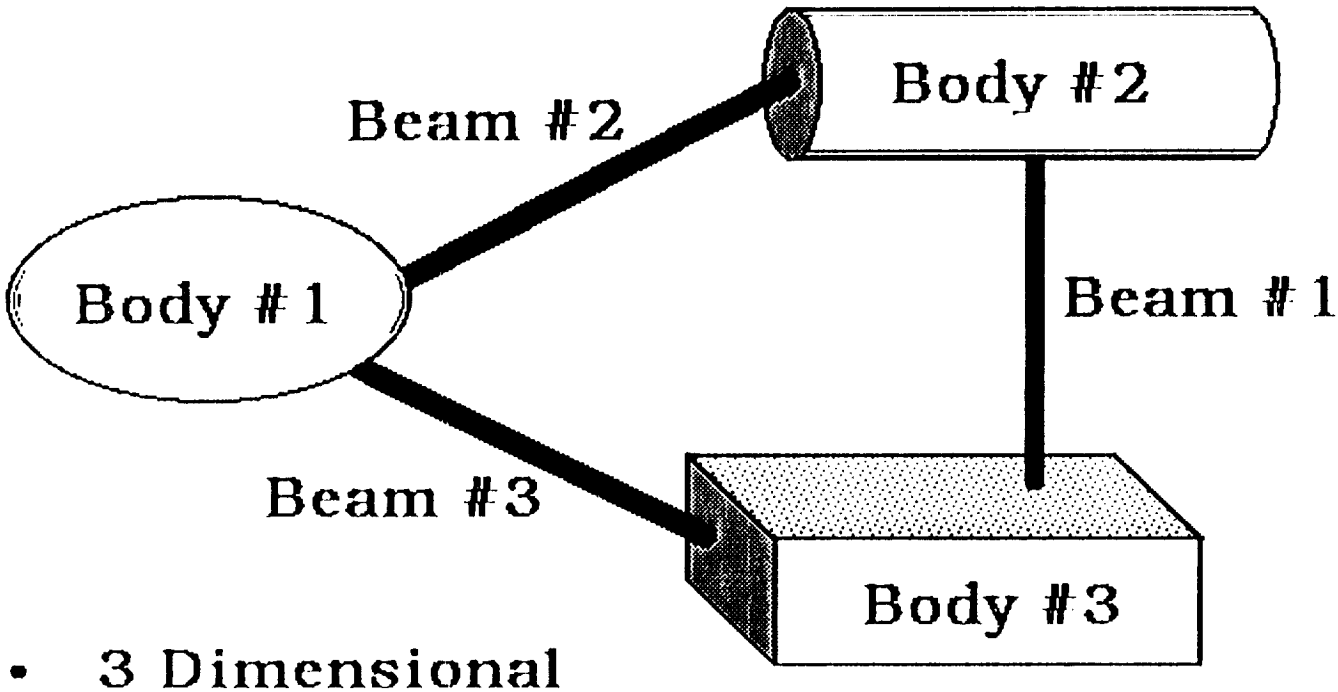
$$M_y = -EI_x u_x''(0) \qquad F_y = -EI_x u_x'''(0)$$

$$M_z = EI_\psi u_\psi'(0) \qquad F_z = EA_z u_z'(0)$$

$$M_{Beam} = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} \qquad F_{Beam} = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}$$

The sign of the forces and moments at the outboard end of the beam are opposite to those of the inboard end.

Generic Configuration



- 3 Dimensional
- Flexible Beams (Bend, Twist, Elongate)
- Rigid Bodies (Full Inertia Matrices)

Partial Differential Equations

Bernoulli-Euler Equations will be used for bending

$$m\ddot{u}_x + EI_x u_x'''' = 0$$

$$m\ddot{u}_y + EI_y u_y'''' = 0$$

String Equations are used for elongation and torsion.

$$m\ddot{u}_z + EA_z u_z'' = 0$$

$$pI_\psi \ddot{u}_\psi + EI_\psi u_\psi'' = 0$$

Examination of each equation will establish the relationship between the "b" parameters and frequency, ϖ.

From $\{ m\ddot{u}_x + EI_x u_x'''' = 0 \}$ we get:

$$m\varpi^2 u_x = b_x^4 EI_x u_x$$

It follows that

$$(b_x L)^2 = \frac{\varpi}{\sqrt{\frac{EI_x}{mL^4}}}$$

Forces and Moments

The forces and moments in body axes are:

$$F_{beam} = P_F \begin{bmatrix} A_x \\ B_x \\ C_x \\ D_x \\ A_y \\ B_y \\ C_y \\ D_y \\ A_z \\ B_z \\ A_\psi \\ B_\psi \end{bmatrix} \qquad M_{beam} = P_M \begin{bmatrix} A_x \\ B_x \\ C_x \\ D_x \\ A_y \\ B_y \\ C_y \\ D_y \\ A_z \\ B_z \\ A_\psi \\ B_\psi \end{bmatrix}$$

Equations of Motion

Force Equations:

$$A_j = Q_{u_j} + T_j^T R_j T_j Q_{u'_j} + \frac{1}{m_j \varpi^2} \sum_{\substack{\text{beam } i \\ \text{body } j}} T_j^T T_{\text{beam } i} P_{F_i}$$

Moment Equations:

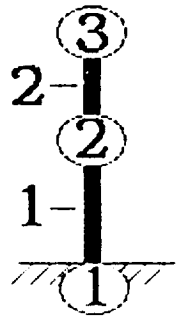
$$A_j = Q_{u'_j} + \frac{T_j^T I_j^{-1}}{\varpi^2} \sum_{\substack{\text{beam } i \\ \text{body } j}} \left(T_{\text{beam } i} P_M + R_{\substack{\text{beam } i \\ \text{body } j}} T_{\substack{\text{beam } i \\ \text{body } j}} P_F \right)$$

Constraint Equations:

$$T_2 u_2(0) - R_2 T_2 u'_2(0) = T_1 u_1(0) - R_1 T_1 u'_1(0)$$

$$T_2 u'_2(0) = T_1 u'_1(0)$$

System Equations



	Beam 1	Beam 2
Body 1	$\frac{\text{Force}(\omega)}{\text{Moment}(\omega)}$	0
Body 2	$\frac{\text{Force}(\omega)}{\text{Moment}(\omega)}$	$\frac{\text{Force}(\omega)}{\text{Moment}(\omega)}$
Body 3	0	$\frac{\text{Force}(\omega)}{\text{Moment}(\omega)}$
Constraint	$\frac{u(L)}{u'(L)}$	$\frac{u(0)}{u'(0)}$

Characteristic Equation:

$$|A(\omega)| = 0$$

Mode Shape Functions

$$u_x = A_x \sin(\beta z) + B_x \cos(\beta z) + C_x \sinh(\beta z) + D_x \cosh(\beta z)$$

$$u = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = Q_u \Theta$$

12 Coefficients form State Vector

Trigonometric and Hyperbolic Functions of ω

Similarly for u', Force, Moment, etc.

Control System

Output feedback control systems can be modeled directly using the structural model.

Collocated Force Control

$$F_{body_i} = K_u u_i + K_{\dot{u}} \dot{u}_i$$

This controller would add to the appropriate term:

$$\Delta A_{i,i} = \frac{1}{m_i} \{K_u + j\omega K_{\dot{u}}\} Q_{u_i}$$

Collocated Moment Control

$$M_{body_i} = K_{u'} u'_i + K_{\dot{u}'} \dot{u}'_i$$

This controller would add to the appropriate term:

$$\Delta A_{i,i} = \bar{I}_i^{-1} \{K_{u'} + j\omega K_{\dot{u}'}\} Q_{u'_i}$$

Higher order compensation for controller can be included as complex functions of ω.

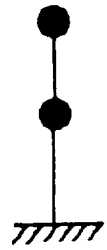
Uncollocated controllers would have the same form but would have different indices to reflect the locations of sensors and actuators.

Dimensionality

Because 12 modal parameters are invlolved for each beam, A like number of equations are involved in the eigen value solution. The SCOLE configuration is an example of such a simple configuration.

12

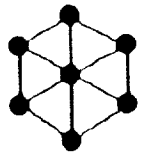
24



The Mini-MAST, a cantilevered beam with lumped masses at the tip and bay 10 will involve 24 equations.

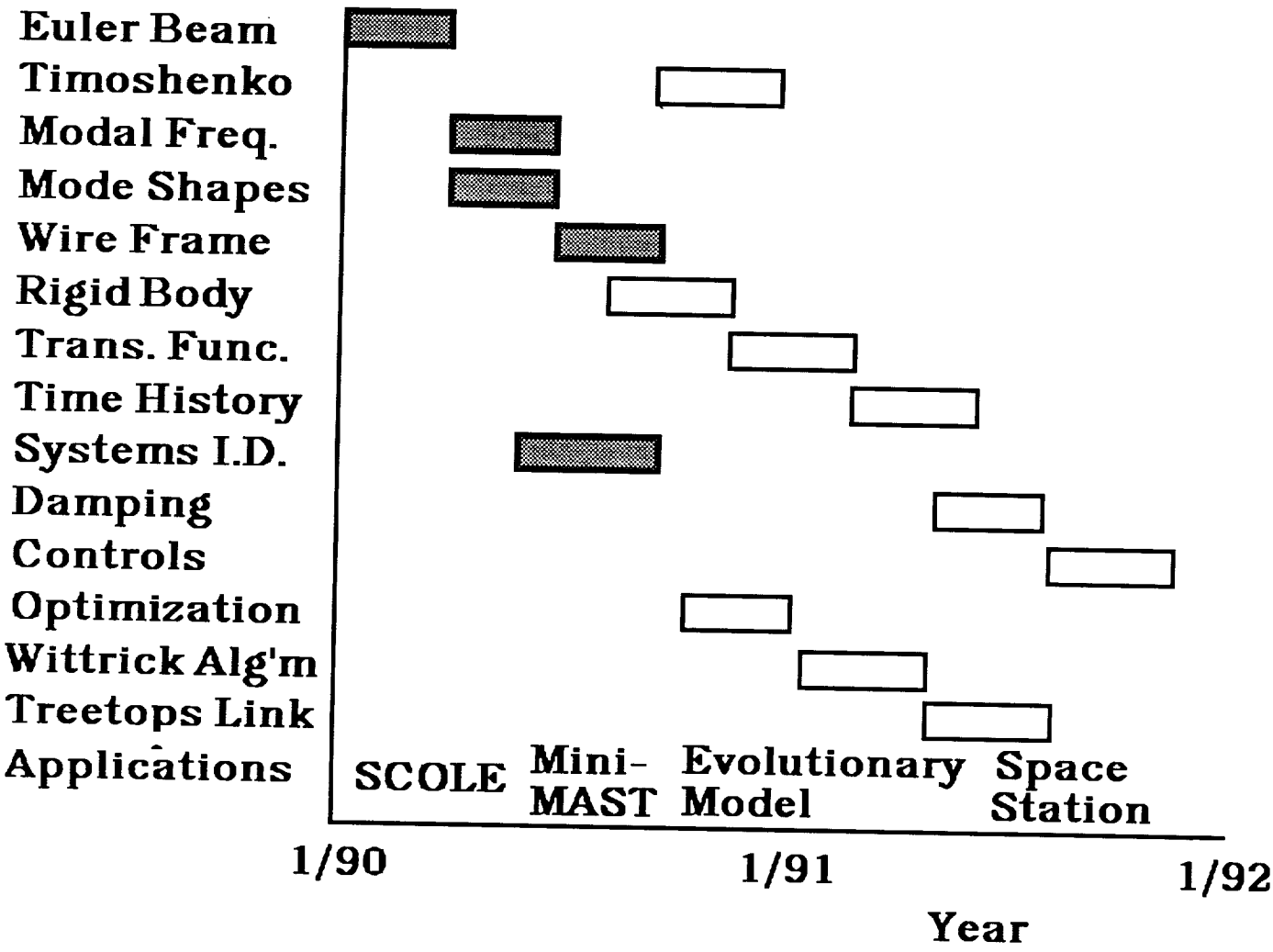
A hexagonal shape with bodies at each point and at its center will involve 144 equations.

144



Clearly, numerical difficulties can be expected to limit the complexity of a configuration that can be handled.

PDEMOD Schedule



PDEMOD Users/Contributors

Augenstine - U. of Maryland

Balakrishnan - U.C.L.A.

Huang/Shen - O.D.U.

Junkins - Texas A. & M.

Kakad - U. of N.C. Charlotte

Kaufman - R.P.I.

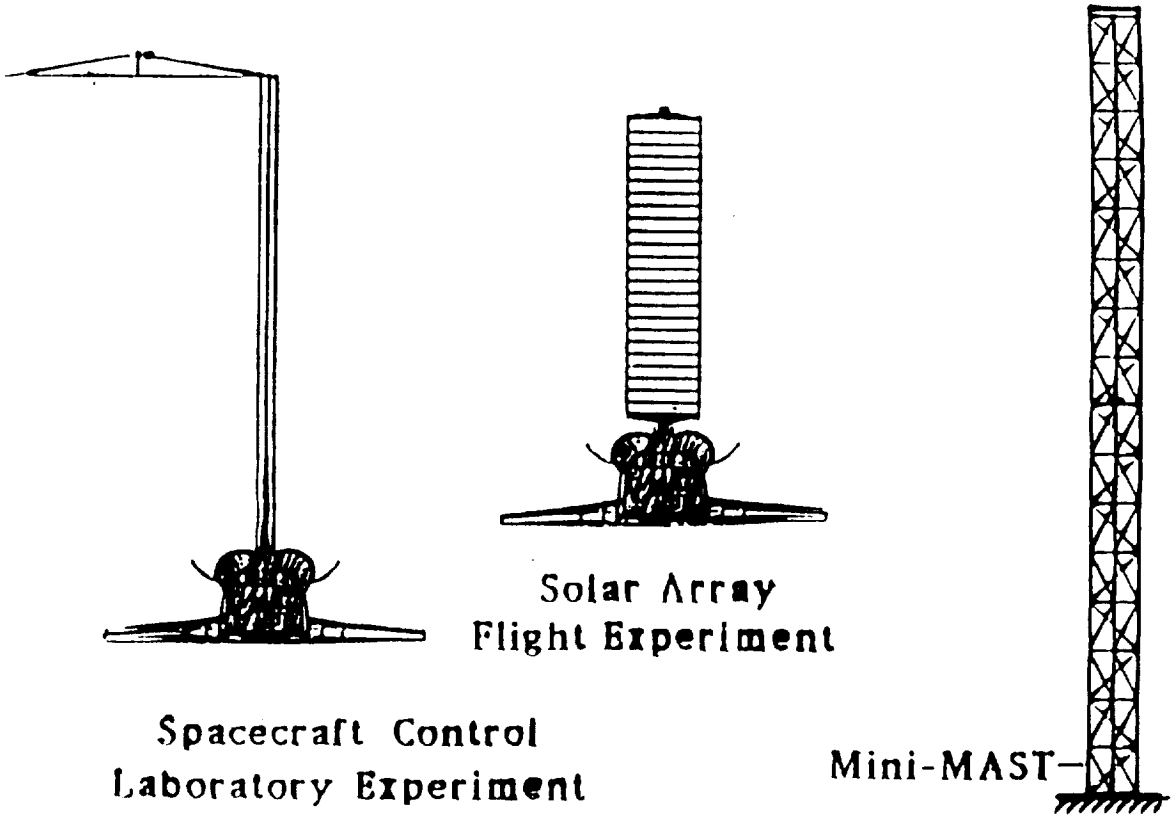
Mitchell - V.P.I. & S.U.

Pitkin - Dynacs Engineering

Williams - U. of Nevada @ Las Vegas

Zimmerman - U. of Florida

Configurations Modeled



Sketches of the Three Spacecraft-Type Structures for which Distributed Parameter Models are Constructed.

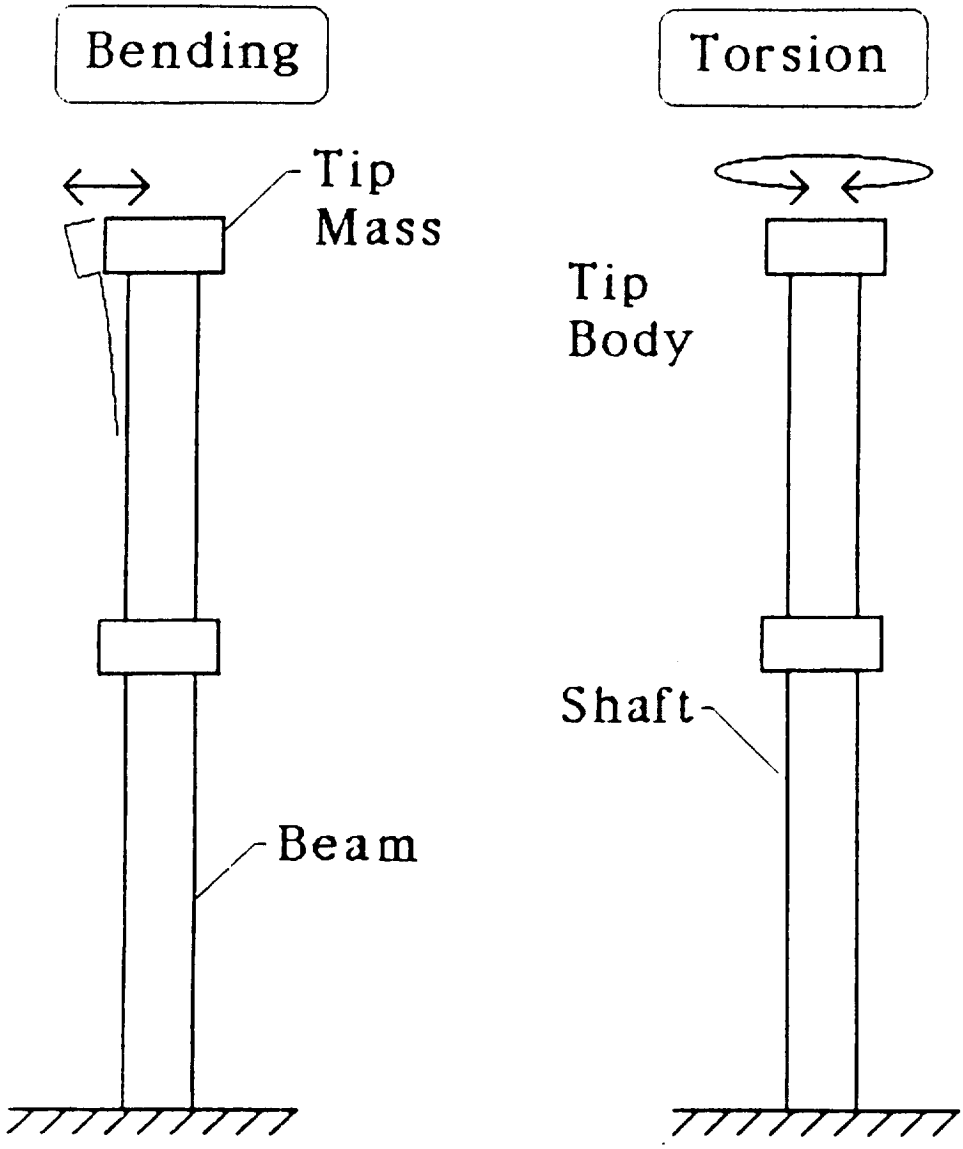


Figure 14. Schematics of Distributed Parameter Models for Bending and Torsion of the Mini-MAST Truss.

Optimization*

$$J = \delta(\Theta)^T M^{-1} \delta(\Theta)$$

$$\frac{\partial J}{\partial \Theta} = 2\frac{\partial \delta}{\partial \Theta}(\Theta)^T M^{-1} \delta(\Theta) + 2\frac{\partial \delta}{\partial \Theta}(\Theta)^T M^{-1} \frac{\partial \delta}{\partial \Theta}(\Theta) \Delta\Theta = 0$$

$$\Delta\Theta = \left[\frac{\partial \delta}{\partial \Theta}(\Theta)^T M^{-1} \frac{\partial \delta}{\partial \Theta}(\Theta)\right]^{-1} \frac{\partial \delta}{\partial \Theta}(\Theta)^T M^{-1} \delta(\Theta)$$

* – Parameter Estimation,

Control Synthesis,

Structural Design

Genetic algorithms (GA's) are one form of directed random search. In GA's, a finite number of candidate solutions are randomly genereated to create an initial population. This initial population is then allowed to evlove overt generations to produce new, and hopefully better designs. The basic motivation behind the development of GA's is that they are robust problem solvers for a wide class of problems. The basic conjecture behind GA's is that evolution is the best compromise between determinism and chance.

GENETIC OPTIMIZATION

- Darwins "Survival of the Fittest"
- Related to Simulated Annealing
- Requires Function Evaluations (no Gradient Information required)
- Seeks to Maximize a "Fitness" Index - related to objective function
- Works with Multiple Designs Simultaneously
    - Identifies "Nearly Optimal" Alternatives
    - Suitable for Parallel Processing

The genetic algorithm was used to determine the optimal continuum beam characteristics of the two sections of the NASA LaRC Mini-Mast for bending in one plane of motion. The objective of the design was to minimize the total structural mass subject to constraints on the tip displacement. As would be expected, the optimizer increases the stiffness of the lower section of the Mini-Mast while decreasing the stiffness of the upper section.

DESIGN PROBLEM

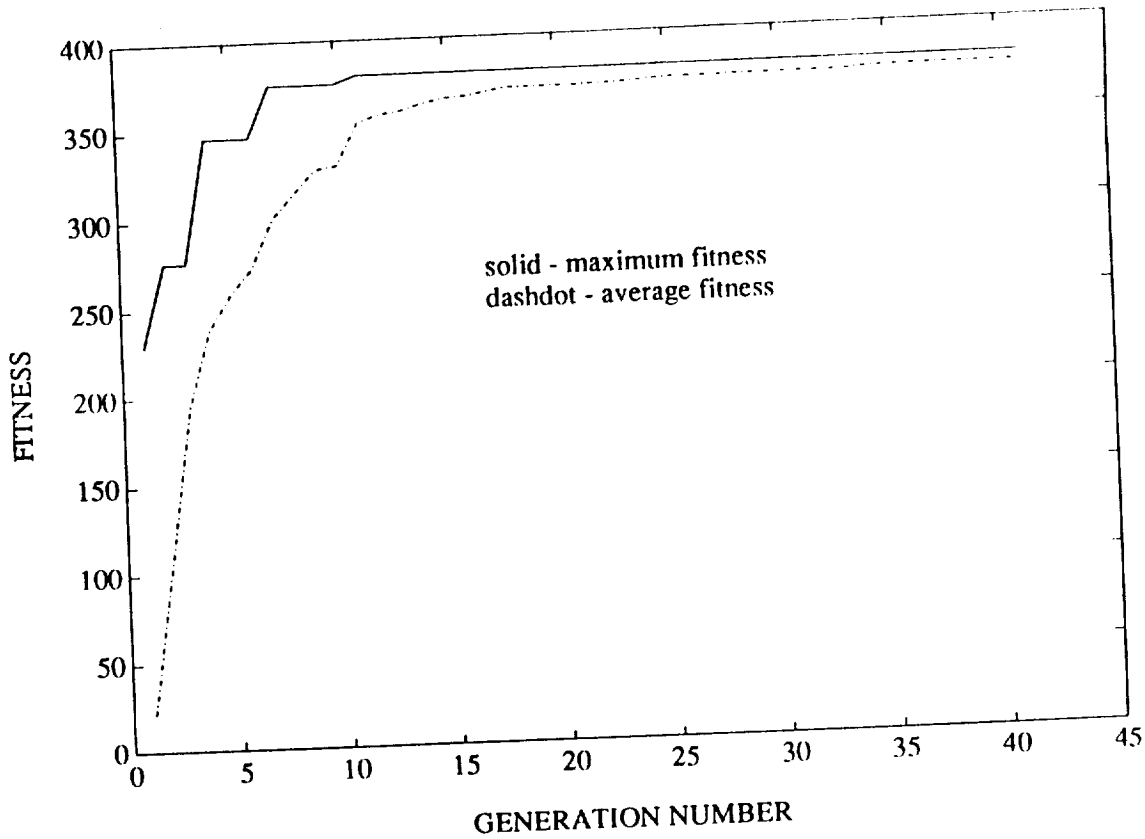
- o FEM Representation of Continuum Model of Mini-Mast (PDEMOD soon to follow)
- o Design Variables - EI(z) of Two Sections ($d*EI(z)_{nominal}$)
- o Problem Statement - Minimize Total Mass Subject to Dynamic Displacement Constraints
- o Results
  - o Nominal - d(1) = d(2) = 1

    Total Mass = 7.1208 slugs
    Constraint Violation = 5.3%
  - o Optimized Design - d(1) = 1.2059 d(2) = 0.4647

    Total Mass = 6.2412 slugs
    Constraint Violation = None

The GA search was run for 40 generations with a population size of 20 members. Thus, the total nuber of function evaluations was 800. The convergence history of the GA is shown below. At a given generation number, the maximum fitness value represents the most fit member in the population whereas the average fitness is the mean fitness of the entire population. It can be seen that the average fitness increases with each new generation, which is a property of the particular GA used. It is also seen that the maximum fitness (i.e. optimal solution) is obtained in an early generation.

CONVERGENCE HISTORY



- Each Generation Represents 20 Function Evaluations (corresponding to 20 Designs)

- Average "Fitness" of the Multiple Designs Increase Quickly

Example Design Problem

Article: Mini-MAST Truss

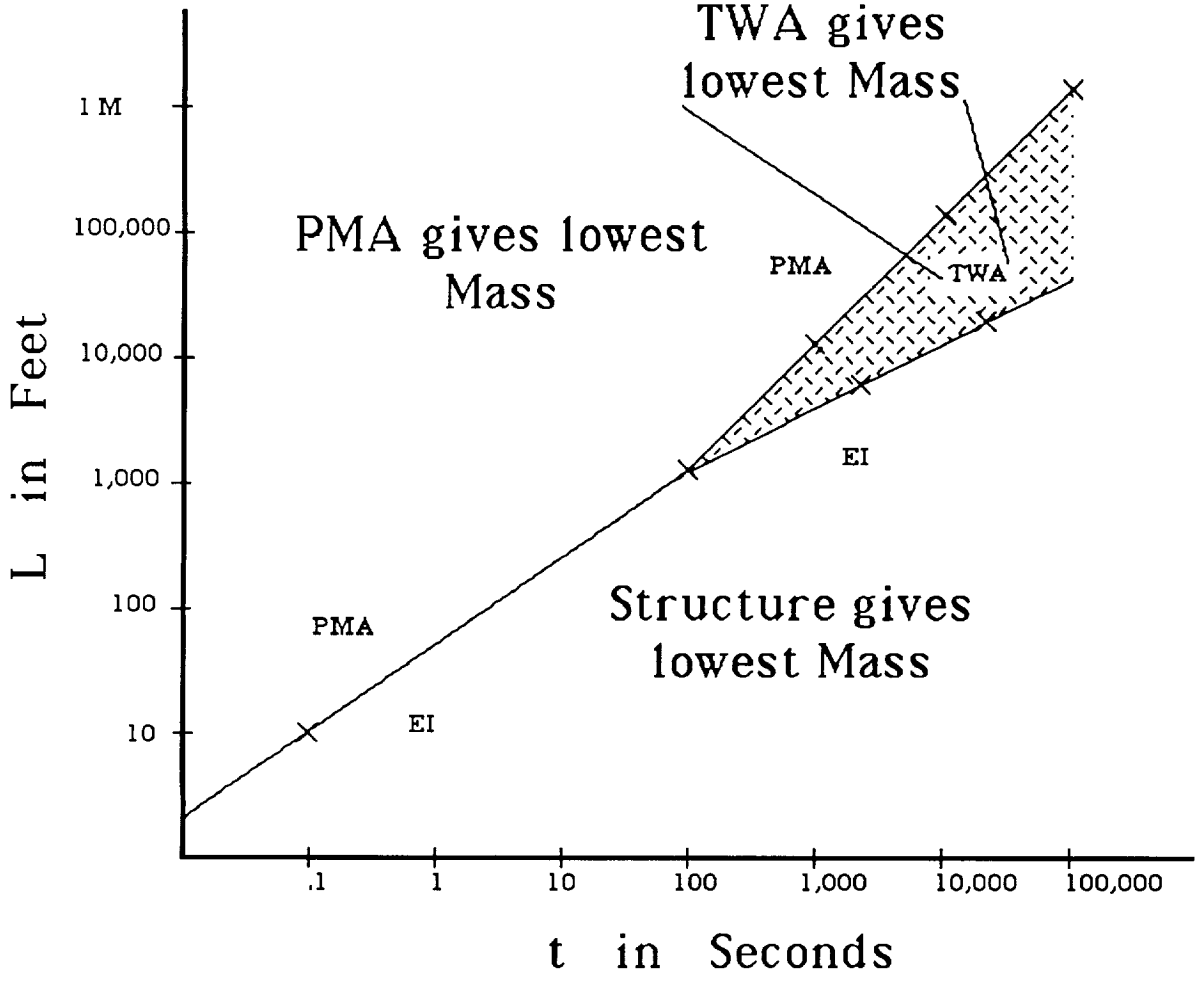
Disturbance: $F = F_{max}$

Specification: $u_{tip} < u_{max}$

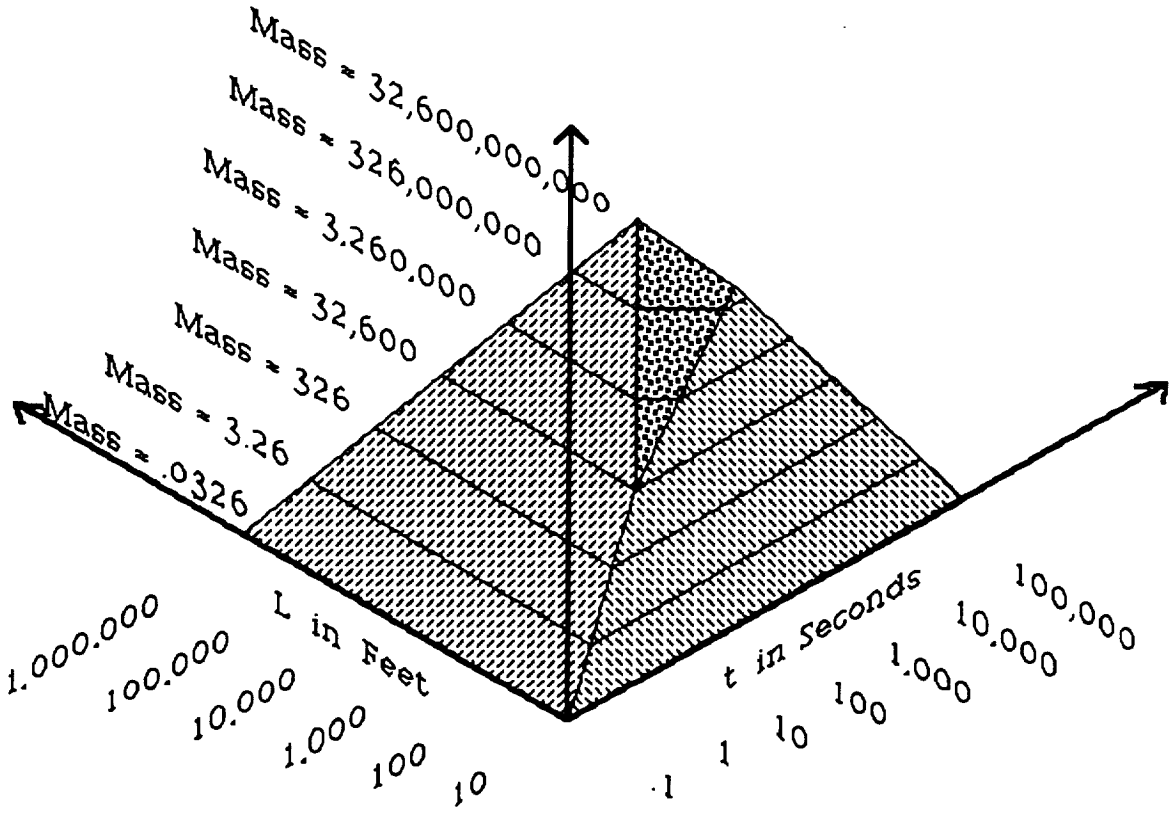
Controls: Proof Mass Actuator
Torque Wheel Actuator

Objective: $J = \min\{m_{truss} + m_{proof} + m_{torque}\}$

Control/Structures Optimization



Control / Structures Optimization



Proof Mass Actuator yields lowest Mass

Torque Wheel Actuator yields lowest Mass

Structure yields lowest Mass

N91-22328

MANEUVER SIMULATIONS OF FLEXIBLE SPACECRAFT BY SOLVING TPBVP

Peter M. Bainum
Distinguished Professor of Aerospace Engineering
Department of Mechanical Engineering
Howard University, Washington D.C.

and

Feiyue Li
Graduate Research Assistant
Department of Mechanical Engineering
Howard University, Washington D.C.

4th NASA Workshop on Computational Control of Flexible Aerospace Systems

Kingsmill Resort, Williamsburg, Virginia
July 11-13, 1990

MANEUVER SIMULATIONS OF FLEXIBLE SPACECRAFT BY SOLVING TPBVP*

Feiyue Li and Peter M. Bainum
Department of Mechanical Engineering
Howard University, Washington D.C.

SUMMARY

The optimal control of large-angle rapid maneuvers and vibrations of a Shuttle-mast-reflector system is considered. The nonlinear equations of motion are formulated by using Lagrange's formula, with the mast modeled as a continuous beam. The nonlinear terms in the equations come from the coupling between the angular velocities, the modal coordinates, and the modal rates. Pontryagin's Maximum Principle is applied to the slewing problem, to derive the necessary conditions for the optimal controls, which are bounded by given saturation levels. The resulting two-point boundary-value problem (TPBVP) is then solved by using the quasilinearization algorithm and the method of particular solutions. In the numerical simulations, the structural parameters and the control limits from the Spacecraft Control Laboratory Experiments (SCOLE) are used.

In the two-dimensional (2-D) case, only the motion in the plane of an Earth orbit or the single-axis slewing motion will be discussed. The effects of the structural offset connection, the so-called axial shortening, and the gravitational torque on the slewing of a flexible spacecraft in Earth orbit is considered. In the case of three-dimensional (3-D) slewing, the mast is modeled as a continuous beam subject to 3-D deformations. The numerical results for both the linearized system and the nonlinear system are presented to compare the differences in their time responses.

INTRODUCTION

Future space missions require large-angle rotational (attitude), 3-dimensional maneuvering ("slew") of a large flexible spacecraft in target acquisition, target tracking, and surveying multiple targets, etc. The whole spacecraft system may be composed of multibodies, including the "rigid" parts involving large relative movements and its flexible parts undergoing "small" deformations. The motions of the system for these space activities are best described by nonlinear equations instead of the "linearized" or linear equations. Many authors have considered the problem of large-angle rapid maneuvers of flexible spacecraft (refs. 1-8). The direct application of Pontryagin's maximum principle to this problem has been applied by many authors (refs. 1-3, 6-8). Most of these researches are concentrated on the 2-D slewing problem, except ref. 8, a recent result for the 3-D slewing of the SCOLE.

2-D Slewings

In ref. 6, the rapid slewing of the 2-D SCOLE has been considered. It is

* Research supported by NASA Grant NSG-1414.

observed that (ref. 6) the time response history of the nonlinear system has a shift from that of the linearized system; the reason for this is mainly due to the structural offset (ref. 7).

The so called axial shortening effect of a beam induced by its transverse displacement has been brought to attention by some authors (Refs. 1-3, 9). Although the shortening terms have been included in the equations (Refs. 1-3), their effect on the slew lacked quantitative analysis; specifically, the numerical examples with and without these terms were not provided. On the other hand, a numerical example in Ref. 9 shows that large differences do result between models with and without the shortening effect. But the numerical example is only for an uncontrolled dynamical response case and the main body's motion is prescribed. In ref. 7, therefore, the shortening terms are considered in the formulation of the equations of motion and numerical examples both with and without these terms are presented to compare the difference between them. Also in ref. 7, the gravitational torque terms are modeled and included in the equations to show their effect on the slewing motion.

3-D Slewings

The direct solution of the open-loop TPBVP for 3-D slews of flexible spacecraft resulted in numerical problems with rank-deficient matrices as stated in ref. 5. However, a different numerical method may be used to overcome this difficulty. In ref. 8, the problem has been solved successfully by using the quasilinearization algorithm and the method of particular solutions for 3-D slews of the asymmetrical flexible SCOLE configuration.

The open-loop slewing approach has several obvious distinct properties. First, the control law is easy to implement in practice for both ground tests and space flight tests. Second, the open-loop solution may serve as a good reference for the feedback control law design, as proposed in refs. 4-5, in which the open-loop solution for a rigid (instead of a flexible) spacecraft is used as the nominal reference trajectory. As an extension to refs. 4-5, it may be helpful if the open-loop solution for the 3-D slew of a flexible spacecraft system could also be used as a nominal reference solution. In addition, through the present study, we can also see how different are the responses of the nonlinear system from those of the linearized system.

In the present report, we will summarize most of the results obtained in refs. 7-8. At the same time, the detailed numerical techniques (briefly mentioned in refs. 7-8) for solving the nonlinear TPBVP will be discussed. Numerical examples are presented to illustrate the use of the techniques and the numerical problems associated with the calculations will be discussed.

EQUATIONS OF THE SYSTEMS

System Configurations

The 2-D and 3-D models of the orbiting SCOLE are shown in figures 1 and 2 , respectively. The Shuttle and the reflector are assumed to be rigid bodies. One end of the flexible mast is fixed to the Shuttle at its mass center, o_s, while the other end is firmly connected to the reflector at an offset point, a_r (x_r in the 2-D

case). Three Euler angles (θ_1, θ_2, θ_3) (θ in the 2-D case) or four quaternions are used to describe the attitude of the Shuttle with respect to an orbital reference system.

The 3-D deformation of the mast consists of two bending deflections $U(z,t)$ and $V(z,t)$ in the x-z and y-z planes, respectively, and torsion $\phi(z,t)$ about the z axis. It is assumed that these deformations are small as compared with the length of the mast and can be expressed by the following modal superposition formulas (ref. 10):

$$U(z,t)=\sum_i \xi_i(z)\alpha_i(t), \quad V(z,t)=\sum_i \eta_i(z)\alpha_i(t), \quad \phi(z,t)=\sum_i \zeta_i(z)\alpha_i(t), \tag{1}$$

where ξ_i, η_i, and ζ_i are modal shape function vector components normalized by a common factor, and α_i is a scaled modal amplitude associated with the ith mode. In the 2-D case, only the first equation is used.

The free vibration of this structure can be considered as a space free-free beam vibration problem with boundary conditions including the masses and moments of inertia of the Shuttle and the reflector. The partial differential equation formulation for this problem (refs. 10-11) can be solved by using the separation of variables method. Note that the natural frequencies and modal functions of the 2-D structure are different from those of the 3-D structure.

2-D Dynamical Equations

After developing the kinetic energy and potential energy, we can obtain the dynamical equations of the 2-D structure in the following matrix form (ref. 7):

$$\left[\begin{array}{c|c} I+2\alpha^T m_a+\alpha^T M_2\alpha & (m_2+M_4\alpha)^T \\ \hline m_2+M_4\alpha & M_3 \end{array}\right]\left[\begin{array}{c} \ddot{\theta} \\ \ddot{\alpha} \end{array}\right]=\left[\begin{array}{c} -2\dot{\theta}\dot{\alpha}^T(m_a+M_2\alpha)-\dot{\alpha}^T M_5\dot{\alpha}-\partial V_g/\partial\theta+Q_\theta \\ \hline -2\dot{\theta}M_a\dot{\alpha}+\dot{\theta}^2(m_a+M_2\alpha)-K\alpha-\partial V_g/\partial\alpha+Q_\alpha \end{array}\right] \tag{2}$$

where α is the n×1 modal amplitude vector, n is the number of flexible modes; M_i(i=2,3,4,5) and M_a are constant n×n matrices; m_a and m_2 are n×1 constant vectors; K is the n×n constant diagonal stiffness matrix; V_g is the gravitational energy of the system which is a function of the orbital rate, ω_0, rotation angle, θ, and the deformation amplitude, $\alpha(t)$; and Q_θ, Q_α are the generalized forces produced by the controls associated with θ and α, respectively.

In equation (2), the elements of the vector, m_a, and the matrices, M_a, M_4, and M_5 have a common factor x_r, the offset. In another words, x_r=0 results in m_a=0, etc. The effect of this structural offset on the slewing is analyzed by changing the value of x_r (from 0 to 32.5 ft). Of course, different x_r imply different natural frequencies and modes. The frequencies decrease as the offset distance increase.

The axial shortening of the beam due to the geometric deformation is also considered by adding a shortening term

$$\Delta z=\frac{1}{2}\int_0^z\left(\frac{\partial U}{\partial z}\right)^2 dz$$

into the formulation. Apparently, Δz is a second order term in the modal amplitude, α. The shortening term is involved in the matrices M_2, M_5 in equation (2).

If we distinguish the terms on the right side of equation (2) according to their order in α and $\dot{\alpha}$, we can see that m_a and M_a are involved in the lower order terms, while M_2 and M_5 contribute to the higher order terms. Therefore, for moderate nonzero values of x_r and small-deformation slews, the influence of the structural offset can be greater than the shortening effect. The linearized equations can be obtained by neglecting all nonlinear terms,

$$\left[\begin{array}{c|c} I & m_2^T \\ \hline m_2 & M_3 \end{array}\right]\left[\begin{array}{c} \ddot{\theta} \\ \hline \ddot{\alpha} \end{array}\right]=\left[\begin{array}{c} 0 \\ \hline -K\alpha \end{array}\right]+\left[\begin{array}{c} -\partial V_g/\partial\theta + Q_\theta \\ \hline -\partial V_g/\partial\alpha + Q_\alpha \end{array}\right]_{LIN} \tag{3}$$

where "LIN" refers to constant and linear terms. Note that on the right side of equation (3), both the structural offset and the shortening terms disappear. This means that by using the linearized equations of motion, we may lose some important information such as the terms associated with the structural offset about the original system.

3-D System Equations

The dynamic equations for the 3-D system can be obtained, by using the Lagrangian method, in the following state form (ref. 8):

$$\dot{y}=\left[\begin{array}{c} \dot{\omega} \\ \hline \dot{\beta} \end{array}\right]=(A+\bar{B}_\alpha)\bar{\bar{\omega}}+(\bar{C}_\beta)\omega+D\alpha+(E+\bar{F}_\alpha)u \tag{4}$$

where α is the $n\times 1$ modal amplitude vector, $\beta=\dot{\alpha}$ is the modal rate vector, ω is the 3×1 angular velocity vector of the Shuttle, $\bar{\bar{\omega}}$ is a 6×1 vector defined by

$$\bar{\bar{\omega}}=\left[\omega_1^2 \quad \omega_2^2 \quad \omega_3^2 \quad \omega_2\omega_3 \quad \omega_3\omega_1 \quad \omega_1\omega_2\right]^T$$

and u is the 9×1 control vector:

$$u=[f_{1x} \; f_{1y} \; f_{1z} \,\vdots\, f_{2x} \; f_{2y} \,\vdots\, f_{3x} \; f_{3y} \,\vdots\, f_{4x} \; f_{4y}]^T.$$

$A_{(3+n)\times 6}$, $D_{(3+n)\times n}$, and $E_{(3+n)\times 9}$, are constant matrices. Also, $(\bar{B}_\alpha)_{(3+n)\times 6}$, $(\bar{C}_\beta)_{(3+n)\times 3}$, $(\bar{F}_\alpha)_{(3+n)\times 9}$, are defined as

$$\bar{B}_\alpha=[B_1\alpha \,\vdots\, B_2\alpha \,\vdots\cdots\vdots\, B_6\alpha], \quad \bar{C}_\beta=[C_1\beta \,\vdots\, C_2\beta \,\vdots\, C_3\beta], \quad \bar{F}_\alpha=[F_1\alpha \,\vdots\, F_2\alpha \,\vdots\cdots\vdots\, F_9\alpha]$$

where B_i, C_i, F_i are all (3+n)×n constant matrices.

Note that in the development of equations (4), a constant inertia (mass) matrix has been assumed for convenience. Meanwhile, all the second and higher order terms of the flexible variables (α and $\dot{\alpha}$) are abandoned in the final equations. But all nonlinear terms ($\bar{\bar{\omega}}$) representing the rigid body motion are retained.

Clearly, the dynamic equations for the rigidized spacecraft can be obtained by deleting all terms related with α and β, that is,

$$\dot{\omega} = \bar{A}\,\bar{\bar{\omega}} + \bar{E}\,u \tag{5}$$

where $\bar{A}$ and $\bar{E}$ are 3×6 and 3×9 constant matrices, respectively.

A linearized form of Equation (4) can also be obtained by deleting all nonlinear terms,

$$\dot{y} = D\alpha + Eu \tag{6}$$

By using the quaternion vector $q=[q_0\ q_1\ q_2\ q_3]^T$, the kinematic equations can be expressed as

$$\dot{q} = \frac{1}{2}\,\tilde{\underline{\omega}}\,q, \qquad \text{where} \quad \tilde{\underline{\omega}} = \begin{bmatrix} 0 & -\omega_1 & -\omega_2 & -\omega_3 \\ \omega_1 & 0 & \omega_3 & -\omega_2 \\ \omega_2 & -\omega_3 & 0 & \omega_1 \\ \omega_3 & \omega_2 & -\omega_1 & 0 \end{bmatrix} \tag{7}$$

OPTIMAL CONTROL-NECESSARY CONDITIONS

For the 3-D slewing problem, we use the following quadratic cost functional,

$$J=\frac{1}{2}\int_0^{t_f}(\alpha^T Q_1\alpha + \omega^T Q_2\omega + \beta^T Q_3\beta + u^T Ru)dt \tag{8}$$

where Q_i, and R are weighting matrices, t_f is the given slewing time. The magnitudes of the controls, u, are bounded,

$$|u_i| \le u_{ib}, \qquad i=1,\cdots,9. \tag{9}$$

The Hamiltonian of the system is,

$$H=\frac{1}{2}(\alpha^T Q_1\alpha + \omega^T Q_2\omega + \beta^T Q_3\beta + u^T Ru + p^T\tilde{\underline{\omega}}q)+\gamma^T\beta+\lambda^T[(A+\bar{B}_\alpha)\bar{\bar{\omega}} + (\bar{C}_\beta)\omega + D\alpha + (E+\bar{F}_\alpha)u]$$

where p, γ, and $\lambda=[\lambda_1\ \lambda_2]^T$ are the costate vectors associated with q, α, ω, and β, respectively. By using the Maximum Principle, the necessary conditions for the unrestricted optimal control problem are the dynamical equations (4, 7) plus the following differential equations for the costates,

$$\dot{p} = -\frac{\partial H}{\partial q} = \frac{1}{2}\tilde{\bar{\omega}}\, p, \qquad \dot{\gamma} = -\frac{\partial H}{\partial \alpha} = -Q_1\alpha - D^T\lambda - (\bar{B}_\alpha^T\lambda)\bar{\bar{\omega}} - (\bar{F}_\alpha^T\lambda)u \tag{10a,b}$$

$$\dot{\lambda}_1 = -\frac{\partial H}{\partial \omega} = -Q_2\omega - \frac{1}{2}[q]p - [\lambda^T(A+\bar{B}_\alpha)]\omega - (\bar{C}_\beta)^T\lambda, \qquad \dot{\lambda}_2 = -\frac{\partial H}{\partial \beta} = -Q_3\beta - \gamma - (\bar{C}_\beta^T\lambda)\omega \tag{10c,d}$$

and the optimal control,

$$\frac{\partial H}{\partial u} = 0, \quad \Longrightarrow \quad u = -R^{-1}(E+\bar{F}_\alpha)^T\lambda \tag{11}$$

The control law (11) is then modified by consideration of the saturations (9):

$$u_i = \left\{ \begin{array}{ll} -u_{ib}, & \text{if } u_{ic} \le -u_{ib} \\ u_{ib}, & \text{if } u_{ic} \ge u_{ib} \end{array} \right\}, \quad \begin{array}{l} \text{otherwise,} \\ u_i = u_{ic} = -[R^{-1}(E+\bar{F}_\alpha)^T\lambda]_i \\ \qquad\qquad i = 1, \ldots, 9. \end{array} \tag{12}$$

The same type of cost functional has been used for the 2-D slew problem. Therefore, the associated necessary conditions are similar to equations (10) and (12).

Two-Point Boundary-Value Problem (TPBVP)

One way of obtaining the optimal control law is to transform the above necessary conditions into the following TPBVP. Let x represent the state vector, and λ represent the costate vector. After substituting the control expressions (12) into equations (4) and (10b), one can obtain two sets of ordinary differential equations for the states and the costates,

$$\dot{x} = f_1(x, \lambda)_{(7+2n)\times 1} \tag{13a}$$

$$\dot{\lambda} = f_2(x, \lambda)_{(7+2n)\times 1} \tag{13b}$$

with the following boundary conditions,

$$x(0) \text{ and } x(t_f) \text{ prescribed}, \quad \lambda(0) \text{ and } \lambda(t_f) \text{ unknown.} \tag{14}$$

Due to the known boundary conditions being specified at the two ends of the slewing time, this problem is usually called the two-point boundary-value problem. This kind of split boundary conditions usually result from the large-angle maneuver requirements, in which the initial (t=0) and final ($t=t_f$) states of the system are specified. By solving this problem, we can obtain the optimal control (based on the necessary conditions). The often used solution strategy is to change the boundary value problem to the initial value problem, i.e., find $\lambda(0)$, the missing initial costates. Once $\lambda(0)$ is obtained, one can solve the equations (13) as an initial value problem by using any numerical integration method. However, owing to the nonlinearity of the equations, there is generally no analytical solution to this problem or simple numerical method to obtain the solution except for some very simple cases such as the linear time-invariant case. The numerical iteration method is the general approach to the this problem.

To start an iteration process, one usually needs an initial guess of $\lambda^{(0)}(0)$. Then, equations (13) or their equivalent form (the linearized version of (13)) are

solved and a $x^{(0)}(t_f)$ is obtained. Based on the difference $\Delta x(t_f)=x^{(0)}(t_f)-x(t_f)$, the correction to $\lambda(0)$, $\Delta\lambda(0)$, is obtained. This gives us a new initial value of $\lambda(0)$, $\lambda^{(1)}(0)$. Hence, the next iteration begins. The iteration process can be terminated when $|\lambda^{(k+1)}(0)-\lambda^{(k)}(0)|$ is less than a given error limit. One can see immediately that if the beginning guess $\lambda^{(0)}(0)$ is close to the true value (converged value) of $\lambda(0)$, the solution will converge and less iterations are needed. However, a "good" guess of $\lambda(0)$ is often difficult to obtain for the general nonlinear problems.

Therefore, the effort for solving the TPBVP is two fold. The first is try to establish a good iteration (correction) method with a wide convergence interval so that it can guarantee convergence even for a "poor" initial guess. The other is try to find a "good" initial guess based on the characteristics of the practical problem and using some simplified mathematical models. In this report, we use the quasilinearization method. We also use the solution of $\lambda(0)$ from the simplified linear, time-invariant model of the system as the initial guess for starting the iteration process.

Linear and Time-Invariant TPBVP

For linear, time-invariant versions of equations (13) (refs. 1-3),

$$\dot{z}=Az, \quad \text{where } z^T=[x^T, \lambda^T] \tag{15}$$

its transition matrix (constant exponential matrix),

$$e^{At_f}=\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

can be used to obtain the initial costates (closed form solution):

$$\lambda(0)=A_{12}^{-1}[x(t_f)-A_{11}x(0)] \tag{16}$$

Nonlinear TPBVP

The continuation (relaxation) process (to increase the participation of the nonlinearity in the solution) and the differential correction (for determination of the initial costate variables) have been used in references 1-3 for the 2-D slewing problem. However, as stated in ref. 5, the extension of these techniques to the 3-D slewing problem has encountered a numerical problem: rank deficiency.

In references 6-8, the quasilinearization method has been successfully used. In this method, one needs to linearize the differential equations (13),

$$\dot{z}=g(z), \quad \text{where } z^T=[x^T, \lambda^T], \quad g^T=[f_1^T, f_2^T] \tag{17}$$

about an approximate solution of this equation in the following form (a series of linearized, time-variant, nonhomogeneous equations):

$$\dot{z}^{(k+1)}=(\partial g/\partial z)z^{(k+1)} + h(z^{(k)}) \tag{18}$$

where $z^{(k)}$ is the kth solution of the same linearized equation. It is also the kth approximate solution of the original nonlinear equations (17). Here, the boundary conditions, (14), are naturally adopted as the boundary conditions of the linearized equations, (18). The control expressions, (11), also need to be linearized (ref. 12):

$$u^{(k+1)}=u^{(k)}-R^{-1}[\bar{F}(\Delta\alpha)]^T\lambda^{(k)}-R^{-1}[E+\bar{F}(\alpha^{(k)})]^T\Delta\lambda \tag{19}$$

where $\Delta\alpha=\alpha^{(k+1)}-\alpha^{(k)}$, and $\Delta\lambda=\lambda^{(k+1)}-\lambda^{(k)}$. By assuming that

$$u^{(k)}=-R^{-1}[E+\bar{F}(\alpha^{(k)})]^T\lambda^{(k)} \tag{20}$$

for the unbounded control case, equation (19) can be rewritten as,

$$u^{(k+1)}=-R^{-1}[\bar{F}(\Delta\alpha)]^T\lambda^{(k)}-R^{-1}[E+\bar{F}(\alpha^{(k)})]^T\lambda^{(k+1)} \tag{21}$$

However, in the bounded control case, equations (12) are considered, that is,

$$u_i^{(k)}=\begin{cases} -u_{ib} \quad \text{or} \quad u_{ib} \\ -\{R^{-1}[E+\bar{F}(\alpha^{(k)})]^T\lambda^{(k)}\}_i \end{cases} \tag{22}$$

Accordingly, at the (k+1)st step, $u^{(k+1)}$ can be determined by

$$u_i^{(k+1)}=\begin{cases} -u_{ib} \quad \text{or} \quad u_{ib}, \quad \text{if} \quad |\{R^{-1}[E+\bar{F}(\alpha^{(k)})]^T\lambda^{(k)}\}_i| \geq u_{ib} \\ -\{R^{-1}[\bar{F}(\Delta\alpha)]^T\lambda^{(k)}-R^{-1}[E+\bar{F}(\alpha^{(k)})]^T\lambda^{(k+1)}\}_i \end{cases} \tag{23}$$

So far, an iteration process is formed. In each iteration, only a linear TPBVP is solved. It is this property that gives this approach the name quasilinearization method.

The linear TPBVP can be solved by many ready-made methods. One of the frequently used algorithms is the method of particular solutions (ref. 13). Let m represent the number of the states (also the costates). Equations (18) can also be rewritten in the following form,

$$\dot{x}(t)=G(t)x(t)+H(t)\lambda(t), \quad \dot{\lambda}(t)=I(t)x(t)+J(t)\lambda(t) \tag{24a,b}$$

From the theorem of the linear system, any solution equation (24a) can be expressed as the linear combination of its m+1 particular solutions, i.e.,

$$x(t)=\sum_{i=1}^{m+1} c_i x^i(t), \quad \text{as long as} \quad \sum_{i=1}^{m+1} c_i=1 \tag{25}$$

where c_i are constants and $x^i(t)$ are the ith particular solution vectors. The method begins by integrating equations (24a, b) forward m+1 times, with the initial conditions,

$$z^1(0)=\begin{bmatrix} x(0) \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix},\ z^2(0)=\begin{bmatrix} x(0) \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix},\ \ldots,\ z^m(0)=\begin{bmatrix} x(0) \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix},\ \text{and}\ z^{m+1}(0)=\begin{bmatrix} x(0) \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

This gives us m+1 particular solutions, $x^1(t)$, $x^2(t)$, $\cdots$, $x^{m+1}(t)$. Substituting these solutions into equations (25), and setting $t=t_f$, we have

$$\sum_{i=1}^{m+1} c_i x^i(t_f)=x(t_f), \quad \sum_{i=1}^{m+1} c_i=1 \tag{26}$$

This is a set of m+1 algebra equations for m+1 unknown constants, c_i. By assuming the existence of inverse of the coefficient matrix, we can obtain the solution, $c=[c_1\ c_2\ \cdots\ c_m]^T$ and c_{m+1}. By doing the following manipulation,

$$z(0)=\sum_{i=1}^{m+1} c_i z^i(0)=\begin{bmatrix} \sum_{i=1}^{m+1} c_i x^i(0) \\ c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix}=\begin{bmatrix} x(0) \\ c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix}$$

one immediately realizes that $c=\lambda(0)$, the missing initial costates.

NUMERICAL RESULTS

2-D Slews

The parameters of the orbiting SCOLE (refs. 10-11) are used. The orbital angular rate is chosen as ω_0=0.001 (rad/s) (orbital altitude h≈ 981 km). The first three natural frequencies used here are:

0.3365257, 2.062547, 5.316669 (hz), for x_r=0;
0.3199540, 1.287843, 4.800169 (hz), for x_r=32.5 (ft) (same as ref. 10)

All simulations are 90 degree rest-to-rest slews and can be represented by:

Case 1 x_r=0, $\bar{u}=u_1$, R=1E-6, t_f=27.6 (s)
Case 2 x_r=0, $\bar{u}=[u_1\ u_2\ u_3\ u_4]^T$, R=DIAG(1E-6, .15, .21, 1E-4), t_f=8.196 (s)
Case 3 x_r=32.5 (ft), $\bar{u}=u_1$, R=1E-6, t_f=27.6 (s)
Case 4 x_r=32.5 (ft), $\bar{u}=[u_1\ u_2\ u_3\ u_4]^T$, R=DIAG(1E-6, .15, .21, 1E-4), t_f=8.196 (s)

Figs. 3a-g display the time histories of $\theta(t)$, $u(L,t)$, $\phi(L,t)$, $\bar{u}(t)$ for Case 4. Clearly, the response of $\theta(t)$ for both linear and nonlinear systems are very close. However, there exist some differences between the two systems in $u(L,t)$, $\phi(L,t)$ and the controls, u. The difference is primarily due to the offset x_r (here, x_r=32.5 ft). When x_r=0, this difference can be reduced markedly, regardless of whether the shortening effect and gravitation are considered. It is also interesting to know that the controls have large differences only around the mid-slew-time.

Table 1 lists the maximum (minimum) values for the displacement, $u(L,t)$, and angle, $\phi(L,t)$, of the beam during the associated slews for all cases. The first line in each case lists the results for the linearized system, while all remaining lines represent those for the nonlinear system with different considerations. For example, ΔL=0 means the shortening effect is not considered. The last column gives the largest relative displacement error, with respect to the linear results, based on

$$(e_{Disp})_* = \text{Max}\left(|MAX_* - MAX_{LIN}|/|MAX_{LIN}|,\ |MIN_* - MIN_{LIN}|/|MIN_{LIN}| \right)$$

Nonlinear System vs. Linearized System First, let us examine line 1 and line 2 in each case. In Case 1, since no offset, the differences between the two lines are very small. In Case 2, where more controllers are used and the slewing time is shortened, the differences increase symmetrically ($|MAX|=|MIN|$), in spite of x_r=0. Case 3 uses the same slewing conditions as used in Case 1, except x_r=32.5 ft. This offset shifts the envelop of the response downward and results in a larger relative displacement error than that in Case 2. Case 4 is the combination of Cases 2 and 3. The shift now is upward which is due to the inclusion of more controllers. When more controllers are used (Ref. 3), the phase of the response reverses, so do the maximum (minimum) amplitudes.

Shortening Effect By comparing line 2 and line 3 in each case, we can see that the shortening terms (1) reduce the amplitude (Cases 2 and 3); (2) increase the amplitude (Case 4); and shift the response upward (Cases 3 and 4). These observations coincide with the fact that Δz only results in a second order effect as compared with the offset effect.

Gravitational Effect By observing lines 3 and 4, we can conclude that the addition of the gravitational torques into the equations of motion has a very small effect on the slew, although they shift the response downward. This is because the orbital rate is much smaller than the slewing rate and the magnitude of the gravitational torque term is much smaller than that of the active control torque term.

3-D Slews

The location of the mass center of the reflector is x_r=18.75 ft, and y_r=32.5 ft. The first five natural frequencies are (hz): .2740493, .3229025, .7487723, 1.244013, 2.051804.

The numerical tests based on the previously described method have been performed for the roll-axis slews, pitch-axis slews, as well as arbitrary-axis slews. All these tests are rest-to-rest slews and the iteration process is terminated after the initial costates are reached within five digit accuracy.

The following procedure is designed to obtain the solution of the nonlinear

TPBVP (figure 4). First, the linear TPBVP based on equation (6) is solved and a nominal trajectory is produced, in which the control is unbounded and the initial costates are calculated by using the transition matrix method. Then, a converged solution for the linear TPBVP with bounded controls is obtained by iterations starting from the previously obtained trajectory. Note that the Euler angles are used in all the above computations.

Next, to obtain the starting solution for the nonlinear TPBVP, the 3 Euler angles and the 3 associated costates are transformed to the 4 quaternions and their 4 costates (from t=0 to t=t_f). Reference 14 provides us the following relationship between the quaternions, q(t) and their costates, p(t):

$$\begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} = \begin{bmatrix} d_0 & -d_1 & -d_2 & -d_3 \\ d_1 & d_0 & -d_3 & d_2 \\ d_2 & d_3 & d_0 & -d_1 \\ d_3 & -d_2 & d_1 & d_0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{27}$$

where d_i are constants. For the case $q(0)=[1\ 0\ 0\ 0]^T$, we can choose $p_0(0)=0$, $d_0=0$. Then,

$$[p_1(0)\ p_2(0)\ p_3(0)]=[d_1\ d_2\ d_3]=d^T \tag{28}$$

The vector d can be determined by

$$d= 2\ [\text{initial Euler angle costate vector}] \tag{29}$$

This result can be proved if one compares the related state and costate equations for both linear and nonlinear TPBVPs (for the case $\omega(0)=0$). After finding the q(t) by using a nonsingular transformation between q(t) and the Euler angles, $\theta_1(t)$, $\theta_2(t)$, and $\theta_3(t)$, one can use equations (27-29) to obtain p(t).

Finally, the nonlinear TPBVP is solved through the quasilinearization process and the method of particular solutions.

Case 3: Fig. 5 shows the results for a simultaneous three-axis slew (θ_1=60, θ_2=30, θ_3=45 deg). The weightings for the states are $Q_1=Q_2=Q_3=0$. In this case, The Shuttle torques (f_{1x}, f_{1y}, and f_{1z}) and the reflector forces (f_{4x}, and f_{4y}) are used. The associated weighting for the control is R=DIAG(1E-4, 1E-4, 1E-4, 0.6, 1.4-3). The slewing time, t_f, is 40 sec. The solid lines in the figures 5a-h responses of the rigidized system, equation (5), while the dotted lines represent the slew results for the flexible spacecraft.

CONCLUSION

Generally, for the 2-D case, the linearized system can predict the system dynamics very well in the slow slewing case. However, in the rapid large-angle slewing problem, the responses of the system deviate noticeably from those described

by the linearized equations if the effects of structural offset and axial shortening are included in the simulations. The structural offset (if any) results in a first order nonlinear effect. The shortening effect results in only a second order nonlinear effect and may not be considered, in the controlled simulations, unless the deformation is out of the linear range. The gravitational effect can be safely neglected in the slew motions considered here.

The application of Pontryagin's Maximum Principle to the large angle slewing maneuver problem has been extended to the slewing of a 3-D flexible spacecraft (SCOLE). A numerical simulation procedure based on the quasilinearization algorithm for solving the resulting nonlinear TPBVP has been established and tested successfully for several examples. The general nonlinear dynamical equations developed here contain all quadratic terms in the angular velocity components of the main body and their coupling with the first order modal amplitudes and modal rates. It is suggested that higher order terms be included if a further analysis is conducted. The numerical results show an important fact that the linearized system can represent the nonlinear system adequately for predicting the major motions but not as well for the secondary motions. For further research, it is recommended that the applicability of this method to more complicated systems be established.

REFERENCES

1. 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, 1980, pp. 578-585.

2. Turner, J.D. and Chun, H.M.: Optimal Distributed Control of a Flexible Spacecraft During a Large-Angle Maneuver. Journal of Guidance, Control, and Dynamics, vol. 7, no. 3, 1984, pp. 257-264.

3. Junkins, J.L. and Turner, J.D.: Optimal Spacecraft Rotational Maneuvers. Elsevier, Amsterdam, 1986.

4. Meirovitch, L. and Quinn, R.D.: Maneuvering and Vibration Control of Flexible Spacecraft. Journal of the Astronautical Sciences, vol. 35, no. 3, 1987, pp. 301-328.

5. Chun, H.M., Turner, J.D. and Juang, J.-N.: Frequency-Shaped Large-Angle Maneuvers. Journal of the Astronautical Sciences, vol. 36, no. 3, 1988, pp. 219-234.

6. Bainum, P.M. and Li, F.: Rapid In-Plane Maneuvering of the Flexible Orbiting SCOLE. AAS/AIAA Astrodynamics Specialist Conf., Stowe, Vermont, Aug. 7-10, 1989, AAS Paper 89-422. to appear Journal of the Astronautical Sciences.

7. Li, F. and Bainum, P.M.: Effects of Structural Offset, Axial Shortening, and Gravitational Torque on the Slewing of a Flexible Spacecraft. 17th International Symposium on Space Technology and Science, Tokyo, Japan, May 20-25, 1990.

8. Bainum, P.M. and Li, F.: Optimal Large Angle Maneuvers of a Flexible Spacecraft. International Conference: Dynamics of Flexible Structures in Space, Cranfield, Bedford, UK, May 15-18, 1990.

9. Kane, T.R., Ryan, R.R., and Banerjee, A.K.: Dynamics of a Cantilever Beam Attached to a Moving Base. Journal of Guidance, Control, and Dynamics, vol. 10, no. 2, 1987, pp. 139-151.

10. Taylor, L.W. and Balakrishnan, A.V.: A Mathematical Problem and a Spacecraft Control Laboratory Experiment (SCOLE) Used to Evaluate Control Laws for Flexible Spacecraft ... NASA/IEEE Design Challenge. Proc. of the 4th VPI&SU Symposium on Dynamics and Control of Large Structures, Blacksburg, VA, June 1983. Revised Jan. 1984.

11. Robertson, D.K.: Three-Dimensional Vibration Analysis of a Uniform Beam with Offset Inertial Masses at the Ends. NASA Technical Memorandum 86393, Sept. 1985.

12. Yeo, B.P., Waldron, K.J. and Goh, B.S.: Optimal Initial Choice of Multipliers in the Quasi-linearization Method for Optimal Control Problems with Bounded Controls. Int. Journal of Control, vol. 20, no. 1, 1974, pp. 17-33.

13. Miele, A. and Iyer, R.R.: General Technique for Solving Nonlinear Two-Point Boundary-Value Problems via the Method of Particular Solutions. Journal of Optimization Theory and Applications, vol. 5, no. 5, 1970, pp. 382-399.

14. Li, F. and Bainum, P.M.: "Numerical Approach for Solving Rigid Spacecraft Minimum Time Attitude Maneuvers. Journal of Guidance, Control, and Dynamics, vol. 13, no. 1, 1990, pp. 38-45.

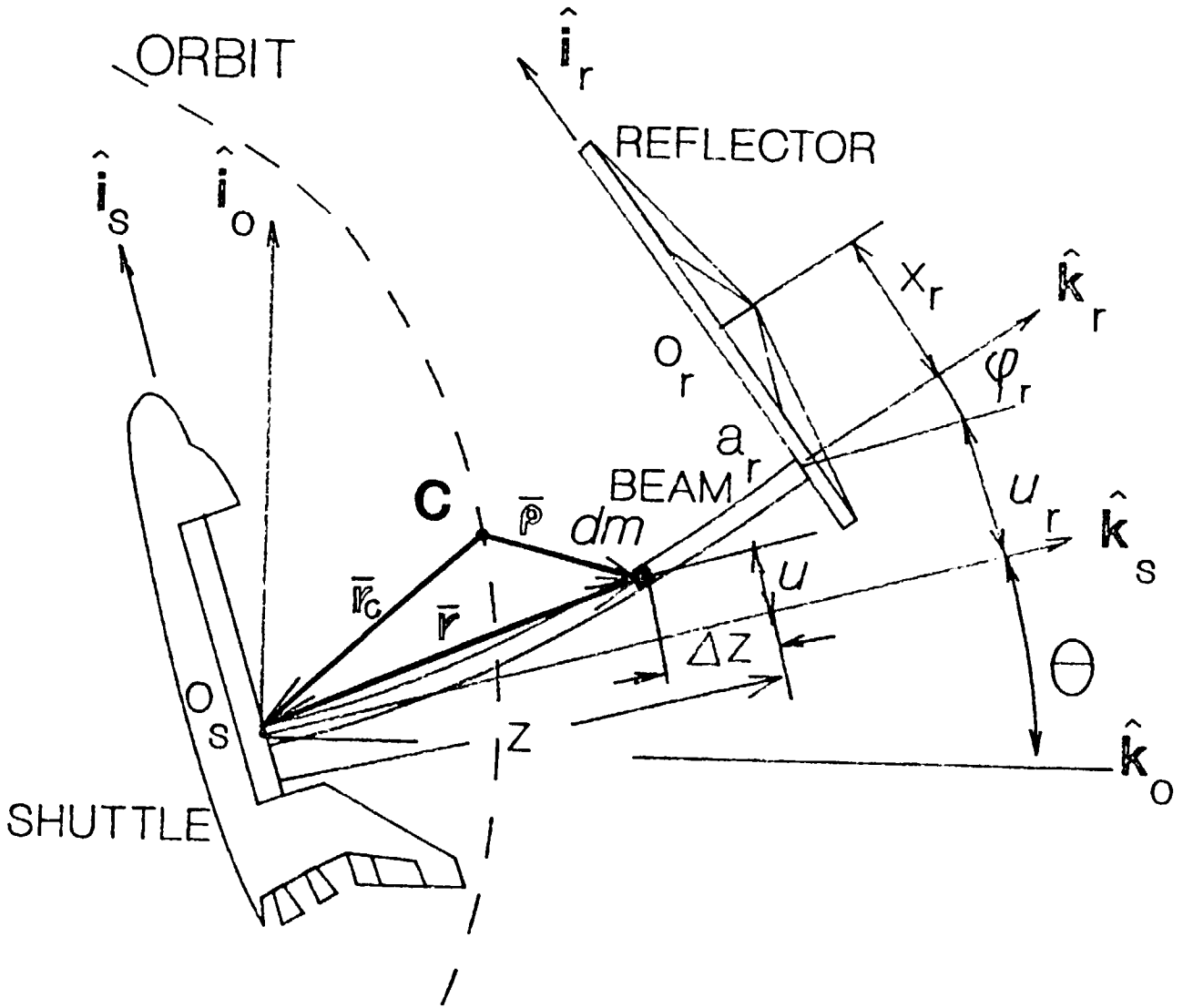


Fig. 1. 2-D Orbiting SCOLE Configuration and Beam Element

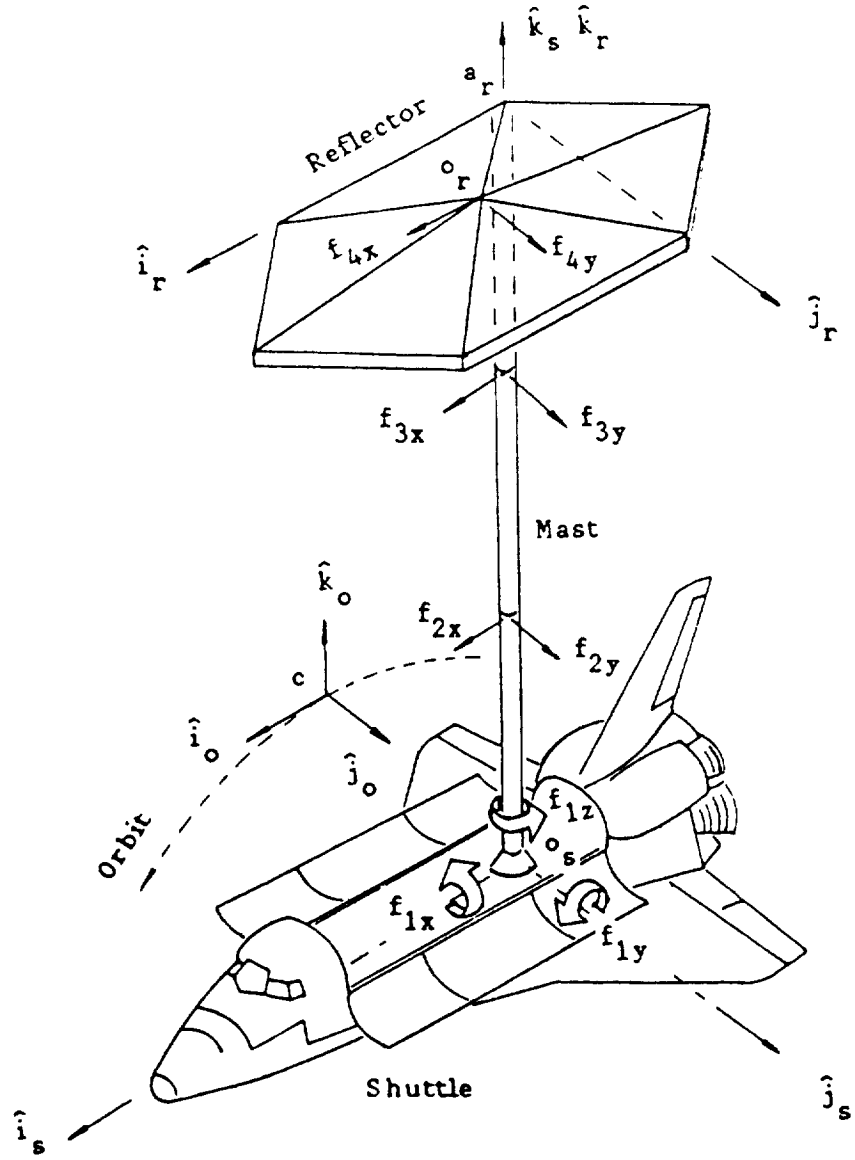


Fig. 2. Drawing of the 3-D Orbiting SCOLE Configuration

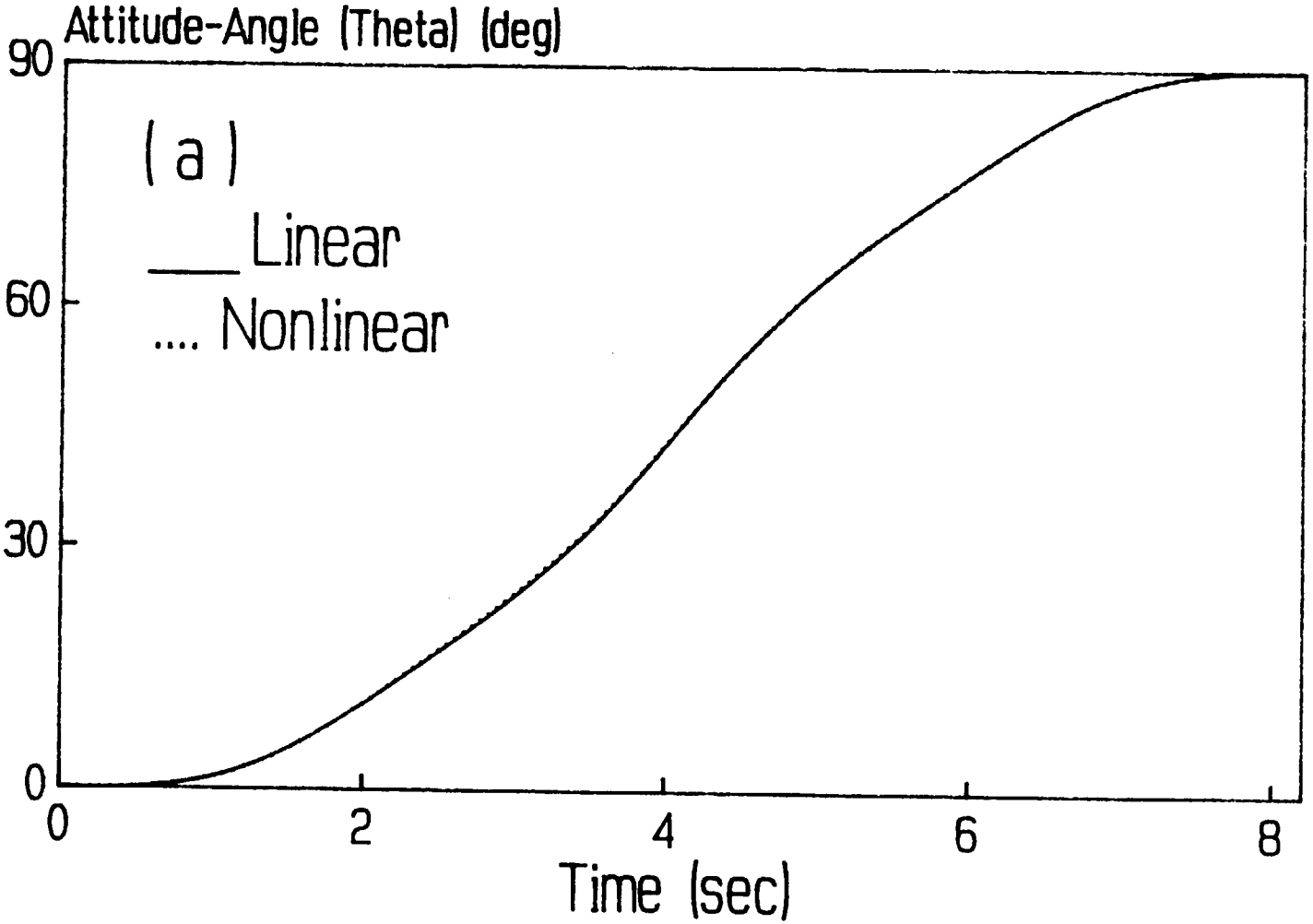


Fig. 3. Results for the 2-D Slew, Case 4: x_r=32.5 ft, $u=[u_1\ u_2\ u_3\ u_4]^T$, t_f=8.196 sec.

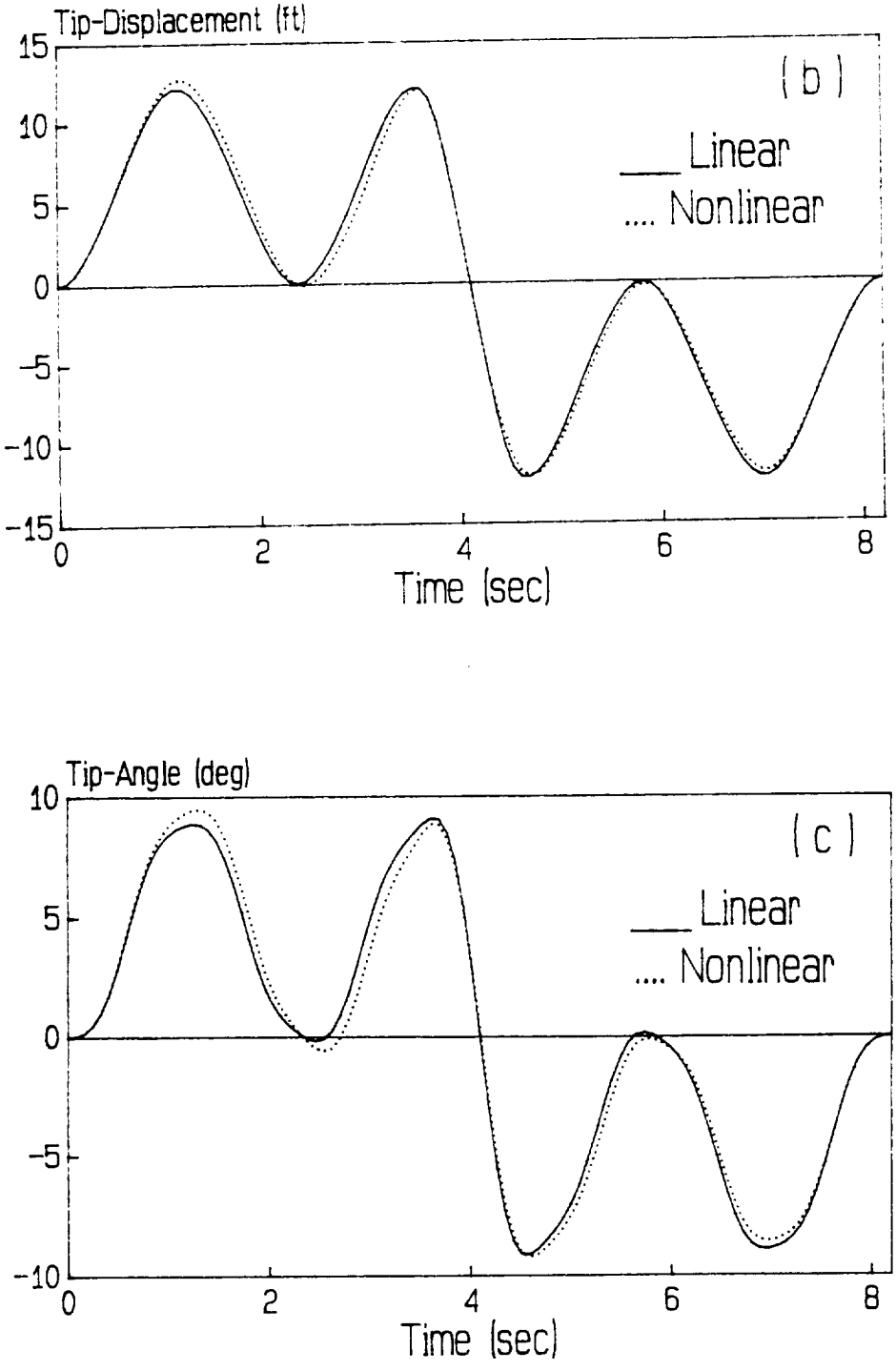
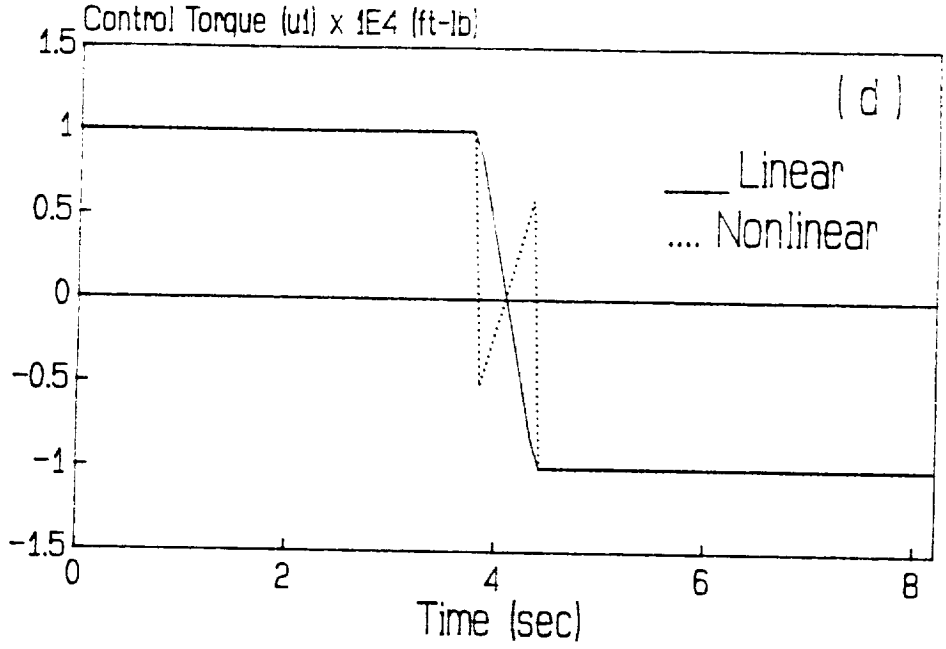


Fig. 3. Results for the 2-D Slew, Case 4: x_r=32.5 ft, $u=[u_1\ u_2\ u_3\ u_4]^T$, t_f=8.196 sec.



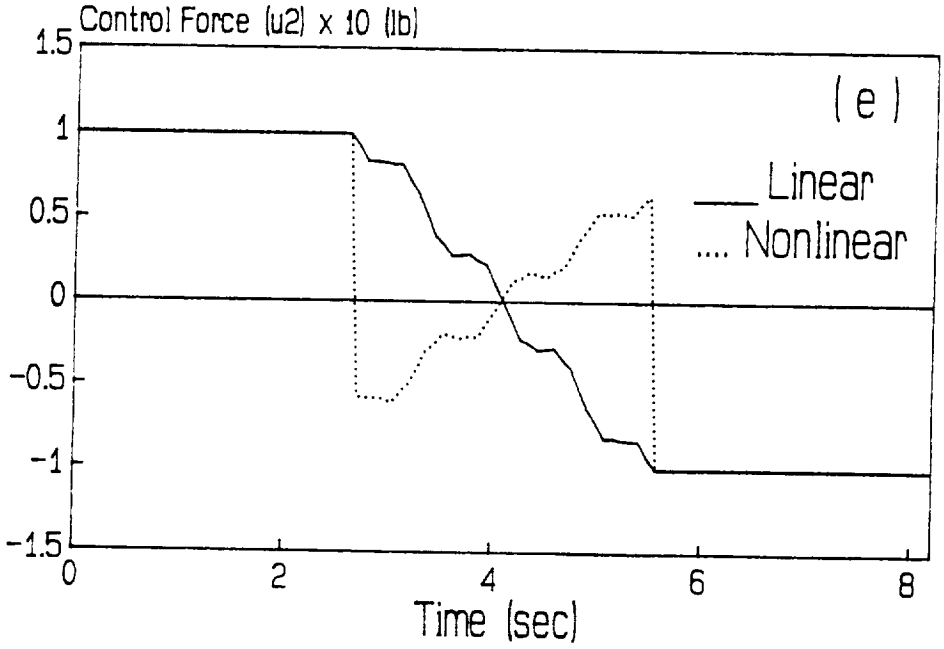
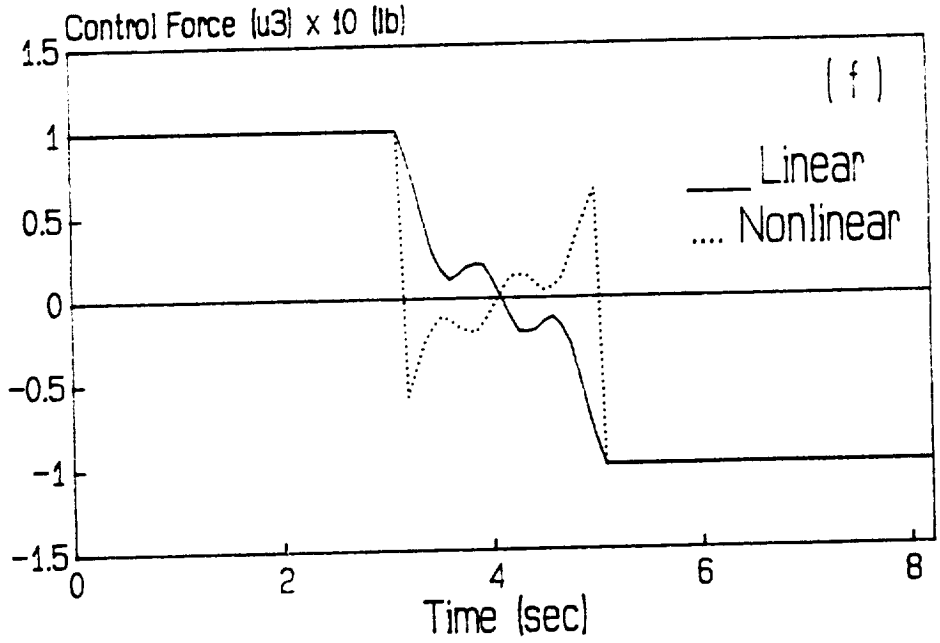


Fig. 3. Results for the 2-D Slew, Case 4: x_r=32.5 ft, $u=[u_1\ u_2\ u_3\ u_4]^T$, t_f=8.196 sec.



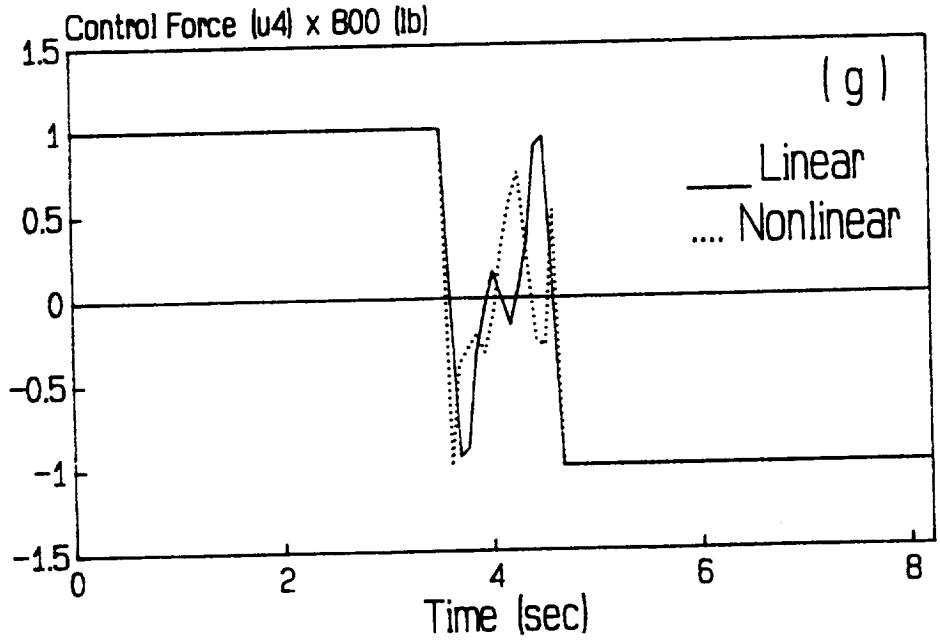


Fig. 3. Results for the 2-D Slew, Case 4: x_r=32.5 ft, $u=[u_1\ u_2\ u_3\ u_4]^T$, t_f=8.196 sec.

Table 1 Tip Displacement and Tip Angle

		Max-Disp (ft)	Min-Disp (ft)	Max-Ang (deg)	Min-Ang (deg)	e-Disp
Case 1	Linearized	0.37727	-0.37727	0.27402	-0.27404	—
$x_r=0$, $\bar{u}=u_1$	$\Delta L=0$, $\omega_0=0$	0.37727	-0.37727	0.27403	-0.27404	0.0%
T=27.6 s	$\Delta L\neq 0$, $\omega_0^0=0$	0.37727	-0.37727	0.27402	-0.27404	0.0%
	$\Delta L\neq 0$, $\omega_0^0\neq 0$	0.37728	-0.37728	0.27403	-0.27405	0.0%
Case 2	Linearized	13.072	-13.072	9.6216	-9.6216	—
$x_r=0$, $\bar{u}=\bar{u}$	$\Delta L=0$, $\omega_0=0$	13.186	-13.186	9.7050	-9.7050	0.87%
T=8.196 s	$\Delta L\neq 0$, $\omega_0^0=0$	13.154	-13.154	9.6847	-9.6847	0.63%
	$\Delta L\neq 0$, $\omega_0^0\neq 0$	13.153	-13.154	9.6842	-9.6845	0.62%
Case 3	Linearized	0.38812	-0.38812	0.30342	-0.30340	—
$x_r\neq 0$, $\bar{u}=u_1$	$\Delta L=0$, $\omega_0=0$	0.38590	-0.40802	0.29110	-0.33940	5.13%
T=27.6 s	$\Delta L\neq 0$, $\omega_0^0=0$	0.38600	-0.40732	0.29118	-0.33884	4.95%
	$\Delta L\neq 0$, $\omega_0^0\neq 0$	0.38586	-0.40803	0.29094	-0.33981	5.13%
Case 4	Linearized	12.191	-12.191	9.1082	-9.1082	—
$x_r\neq 0$, $\bar{u}=\bar{u}$	$\Delta L=0$, $\omega_0=0$	12.734	-12.061	9.4541	-9.1299	4.45%
T=8.196 s	$\Delta L\neq 0$, $\omega_0^0=0$	12.796	-12.052	9.5067	-9.2030	4.96%
	$\Delta L\neq 0$, $\omega_0^0\neq 0$	12.795	-12.054	9.5052	-9.2061	4.95%

SOLUTION PROCEDURE FOR NONLINEAR TPBVP

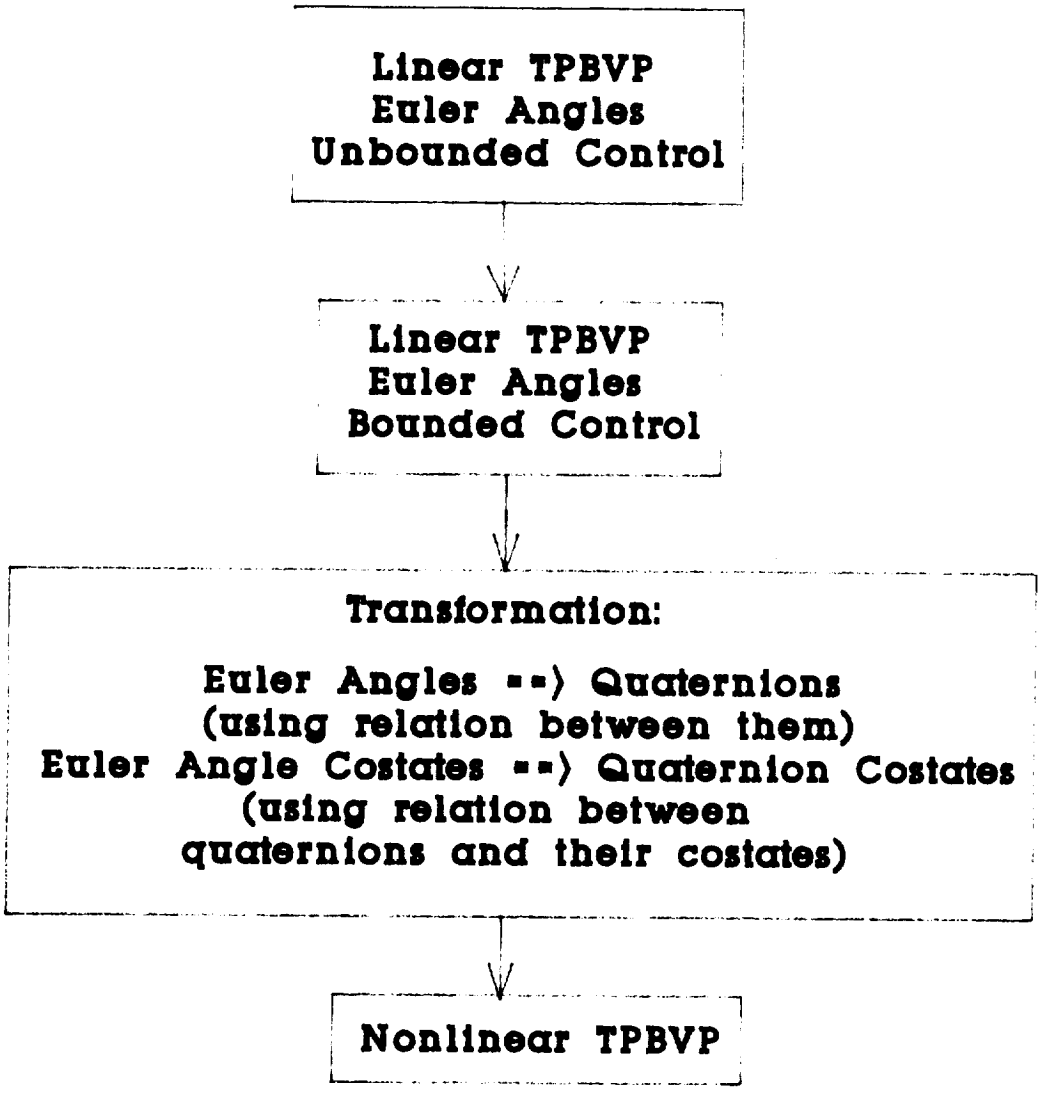


Fig. 4. Solution Procedure for Nonlinear TPBVP

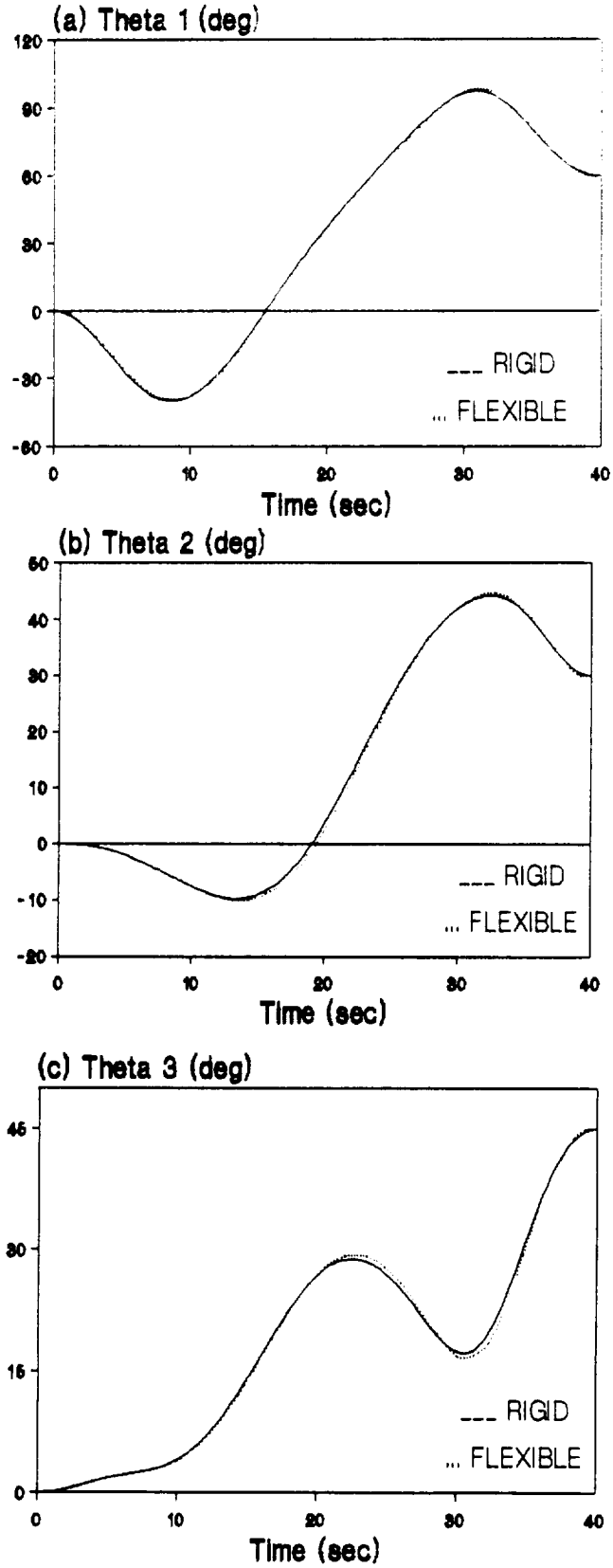


Fig. 5. Results for the 3-D slew, Case 3: 60-30-45 deg-Slew, Shuttle Torques + Reflector Forces, R=Diag(1E-4, 1E-4, 1E-4, 0.6, 1.4E-3).

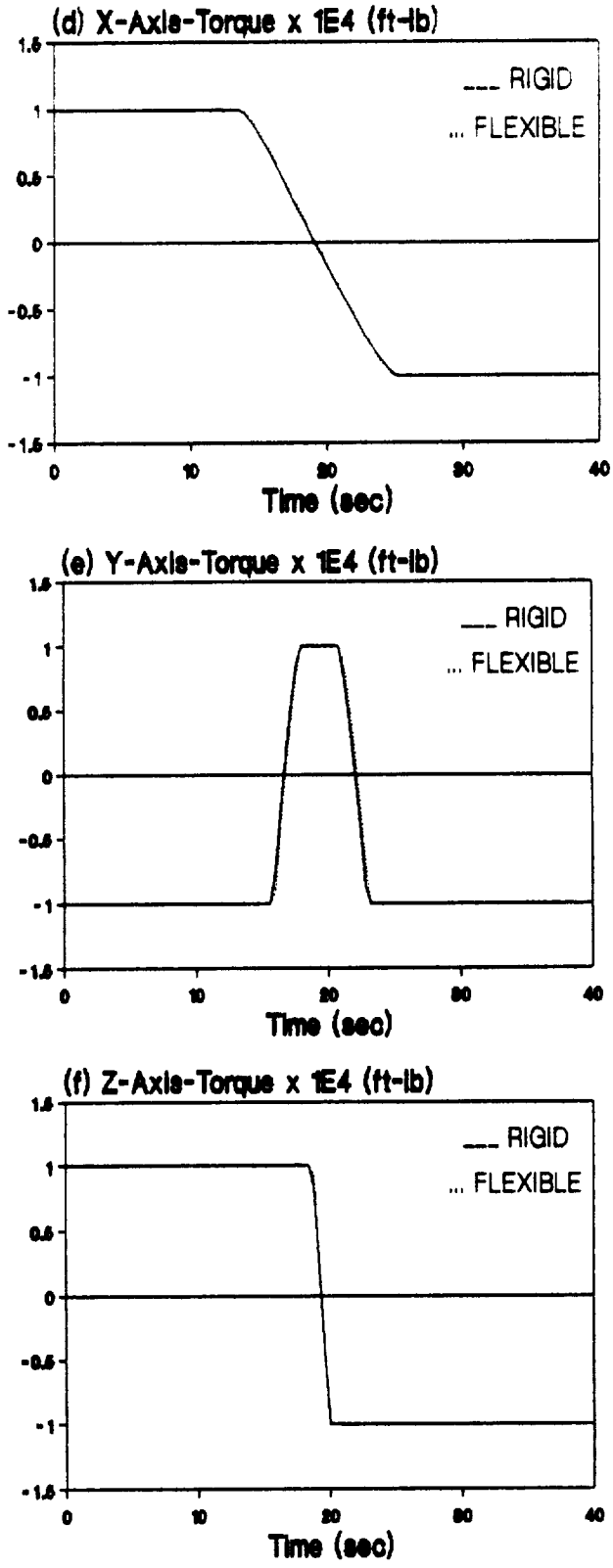
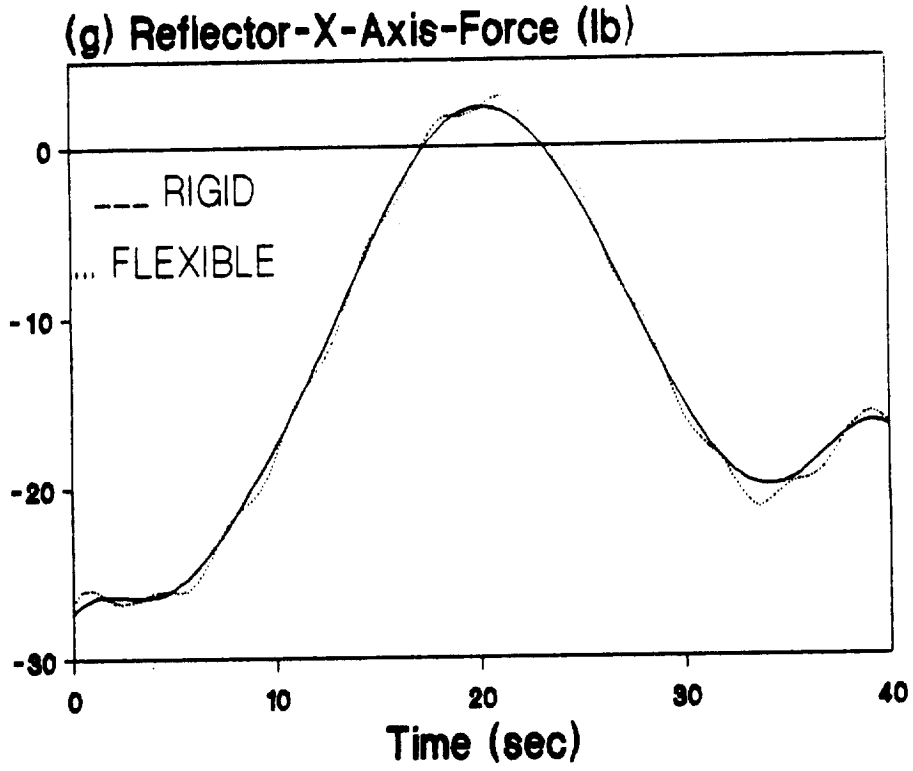


Fig. 5. Results for the 3-D slew, Case 3: 60-30-45 deg-Slew, Shuttle Torques + Reflector Forces, R=Diag(1E-4, 1E-4, 1E-4, 0.6, 1.4E-3).



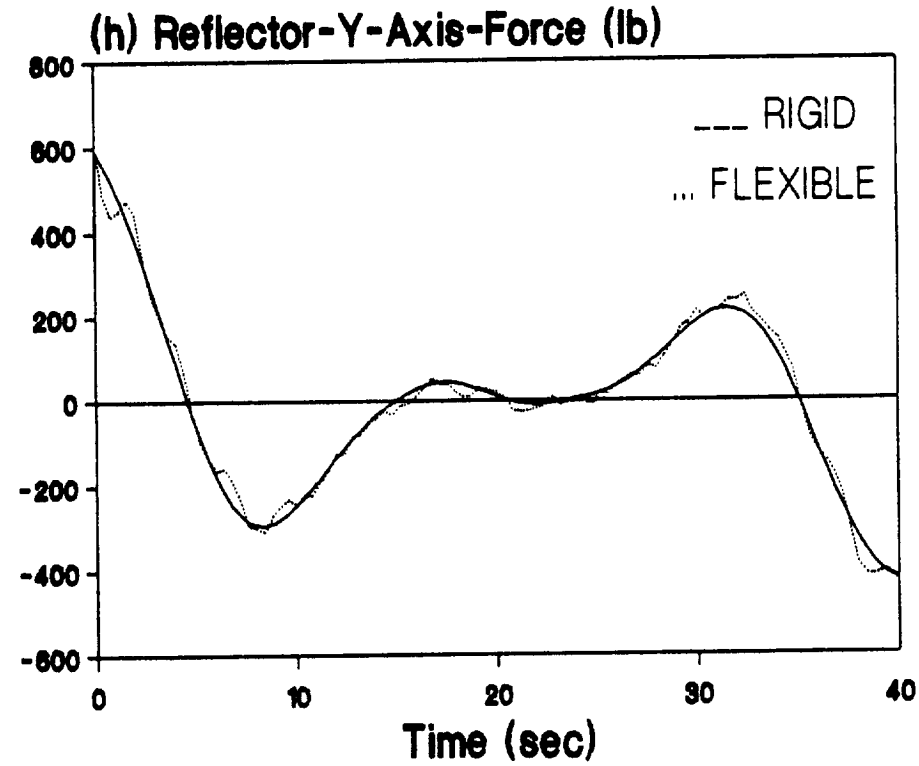
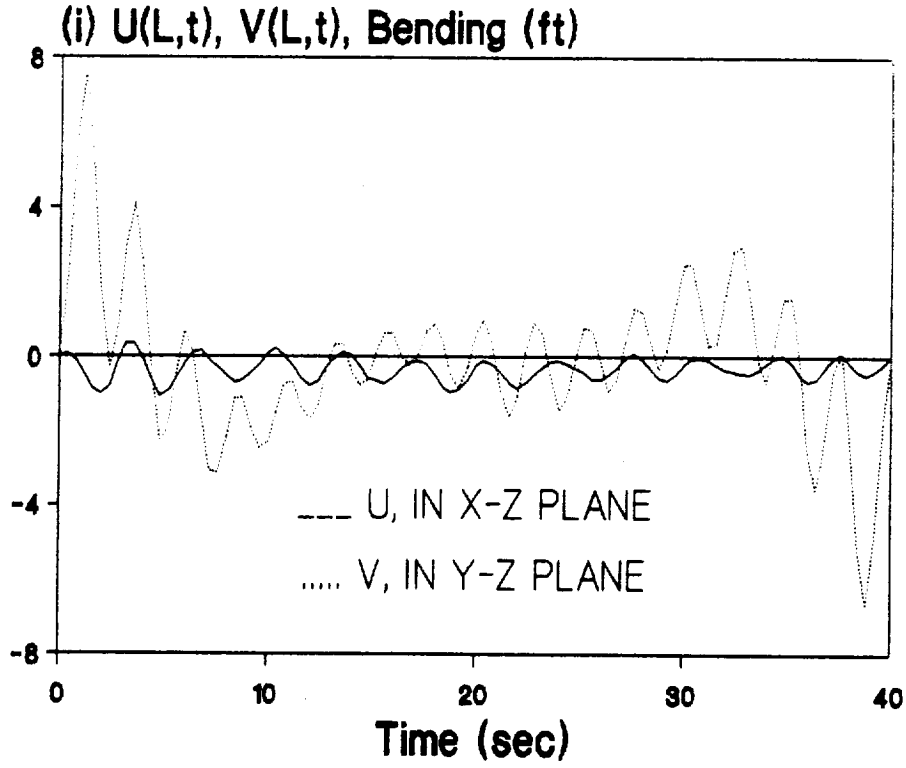


Fig. 5. Results for the 3-D slew, Case 3: 60-30-45 deg-Slew, Shuttle Torques + Reflector Forces, R=Diag(1E-4, 1E-4, 1E-4, 0.6, 1.4E-3).



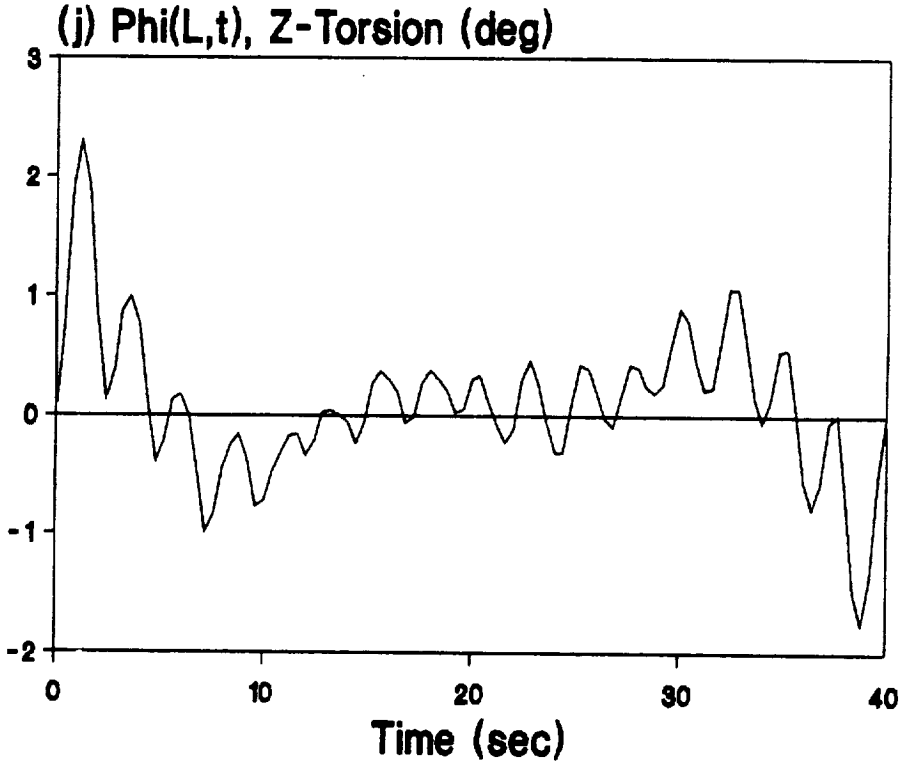


Fig. 5. Results for the 3-D slew, Case 3: 60-30-45 deg-Slew Shuttle Torques + Reflector Forces, R=Diag(1E-4, 1E-4, 1E-4, 0.6, 1.4E-3).

CONTROL EFFORT ASSOCIATED WITH MODEL REFERENCE ADAPTIVE CONTROL FOR VIBRATION DAMPING

Richard Scott Messer and Raphael T. Haftka
Department Of Aerospace and Ocean Engineering
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24060

SUMMARY

The performance of Model Reference Adaptive Control (MRAC) is studied in numerical simulations with the objective of understanding the effects of differences between the plant and the reference model. MRAC is applied to two structural systems with adjustable error between the reference model and the actual plant. Performance indices relating to control effort and response characteristics are monitored in order to determine what effects small errors have on the control effort and performance of the two systems. It is shown that reasonable amounts of error in the reference model can cause dramatic increases in both the control effort and response magnitude (as measured by energy integrals) of the plant

INTRODUCTION

During the past decade, researchers have shown much interest in control and identification of large flexible structures, with emphasis on Large Space Structure (LSS). Furthermore, our inability to model these large structural systems accurately has generated extensive research into adaptive controllers capable of maintaining stability in the face of large structural uncertainties as well as changing structural characteristics. However, most of this research has been strictly theoretical in nature (e.g., refs. 1-10) and experimental verification (e.g., refs. 11,12) of the proposed theories is lagging far behind. In addition, the focus of most theoretical research has been on designing stable adaptive controllers with little or no concern for the issue of control effort.

While it is possible to design an adaptive controller that will stabilize a structure even if we have a very poor model, the control effort may be very high. The objective of the present paper is to study the correlation between the control effort and the fidelity of the structural model. Specifically, the first step is to demonstrate that the effort associated with an adaptive control system is sensitive to knowledge of the structure. For this purpose the popular Model Reference Adaptive Control (MRAC) method was selected and two examples were studied in detail. In this paper, we monitor four performance indices: Maximum Control Force, Quadratic Control Effort,

Kinetic Energy, and Potential Energy. These performance indices allow us to evaluate the effects of errors in the theoretical model. Numerical simulations were used to see how each performance index changed when errors were introduced into the system. Section II summarizes the MRAC algorithm, section III shows how a simply supported beam can be sensitive to the choice of Reference Models, section IV presents the sensitivity of a more complicated structure and section V provides concluding remarks.

MODEL REFERENCE ADAPTIVE CONTROL

Adaptive controllers generally fall into two classifications, direct and indirect. The basic difference between the two classifications is system identification. Indirect adaptive methods (e.g., refs. 9-10, 13-14) require system identification before the adaptive gains in the controller can be updated, whereas direct methods (refs. 1-8, 11-12) do not use system identification. MRAC is one of the more popular direct methods (refs. 1-7). MRAC methods adaptively tune the controller gains, forcing the actual system to follow some ideal reference model. Because this reference model can be of lower order than a typical model of the actual system, this method is very attractive for applications to LSS, where structural models can be of very high order and require truncation for use with any controller. Figure 1 shows a block diagram of a generalized MRAC system (ref. 6).

PROBLEM FORMULATION

The LSS, or controlled plant can be represented in standard state space form:

$$\dot{X}_p(t) = A_p X_p(t) + B_p U_p(t) \tag{1a}$$

$$Y_p(t) = C_p X_p(t) \tag{1b}$$

where $X_p \in R^{Np}$, $U_p \in R^M$, $Y_p \in R^M$ and A_p, B_p, C_p are of appropriate dimensions. It is assumed that (A_p, B_p) is controllable, (A_p, C_p) is observable, and that the number of inputs (M) is equal to the number of outputs.

A stable reference model which specifies the desired performance of the plant is also described by a state space representation,

$$\dot{X}_m(t) = A_m X_m(t) + B_m U_m(t) \tag{2a}$$

$$Y_m = C_m X_m(t) \tag{2b}$$

where $X_m \in R^{Nm}$, $U_m \in R^M$, $Y_m \in R^M$ and A_m, B_m, C_m are of appropriate dimensions. For practical application to LSS the following condition must be true

$$N_p >> N_m \tag{3}$$

To aid in measuring how close the actual plant is to the reference model, the output error between the plant and the reference model is defined as

$$e_y(t) = Y_m(t) - Y_p(t) \tag{4}$$

Since the output error tells us how close the actual plant is to the desired performance of the reference model, the objective of any adaptive update scheme is to design a control input which forces the output error to zero asymptotically.

ADAPTIVE LAW

The adaptive mechanism used in this paper is based on the work of Sobel, Kaufman and Mabius (ref. 1) and has no provisions for the destabilizing effects of noise. However, it has found important application to LSS. The control input is written as,

$$U_p(t) = K(t)\, r(t) \tag{5}$$

where

$$r^T = \left[e_y^T, X_m^T, U_m^T\right] \tag{6a}$$

$$K(t) = \left[K_e(t), K_x(t), K_u(t)\right] \tag{6b}$$

The adaptive gain K(t) is calculated as the sum of a proportional component $K_{pr}(t)$ and an integral component $K_I(t)$ so that

$$K(t) = K_{pr}(t) + K_I(t) \tag{7}$$

The adaptive laws for $K_{pr}(t)$ and $K_I(t)$ are given as

$$K_{pr}(t) = e_y r^T T^* \tag{8}$$

$$\dot{K}_I(t) = e_y r^T T \tag{9}$$

where T^* and T are time invariant weighting matrices of appropriate dimension chosen by the designer. Sufficient conditions for global stability are presented in (ref 1-3, 11-12) and will only be summarized here.

1. $T \;\&\; T^* > 0$
2. there exist $P = P^T > 0$ and $Q = Q^T > 0$ such that

$$PB_p = C_p^T$$

$$PA_p + A_p^T P = -Q$$

Condition 1 is met simply by choosing appropriate matrices (i.e., the identity matrix). Condition 2 is equivalent to the assumption that the open-loop plant transfer function matrix

$$Z(s) = C_p(sI - A_p)^{-1}B_p \tag{10}$$

is strictly positive real. This condition is met for any LSS having small but non-zero inherent damping and colocated sensors and actuators.

CONTROL EFFORT

In order to assess the added implementation costs of MRAC in systems where reasonable amounts of error would occur, we have adopted the following procedure. The first step is to choose a linear system to represent the actual physical system. Next, we create a reference model which specifies the desired performance and has some measurable amount of error. Previous examples, see (ref. 2), have chosen the reference model to be a reduced model of the actual plant with the same frequencies and mode shapes plus extra damping. While this would be the ideal situation, it is not probable that we would have an exact theoretical model. For this reason we have intentionally introduced errors between our reference models and the actual plant model. To aid in quantifying the increased effort due to the errors, we calculate the following performance indices:

the maximum control force required by each actuator,

$$U_i^{max} = Max(\mid U_i(t) \mid) \quad 0 \le t \le t_{final}, \quad i = 1,\ldots,M \tag{11}$$

the quadratic control effort,

$$U_{total} = \int U^T U dt \tag{12}$$

the integral of the potential energy of the system,

$$PE = \frac{1}{2}\int X^T K X dt \tag{13}$$

and the integral of the kinetic energy

$$KE = \frac{1}{2}\int \dot{X}^T M \dot{X} dt \tag{14}$$

where the first two performance indices measure the control effort and the second two provide information about system response characteristics. These performance indices allow us to see how increments of error affect the cost and performance of the system.

SIMPLY SUPPORTED BEAM EXAMPLE

The first example is a simply supported beam with a variable concentrated mass at the mid-span and a velocity sensor and force actuator colocated at one-sixth span (see figure 2). This simple structure is similar to a structure used by Bar-Kana, Kaufman & Balas (ref. 2) for demonstrating the MRAC method. The only difference between the present structure and the structure of reference 2 is the variable concentrated mass. The variable concentrated mass at the mid-span was used to create error in the system due to unknown mass characteristics. The concentrated mass was varied between 0-20% of the mass of the beam, with zero mass corresponding to an exact reference model. It should be noted that the reference model was held constant while the plant model was varied to match changes in the concentrated mass.

The beam was modeled with 12 beam finite elements with a displacement and rotational Degree of Freedom (DOF) at each nodal point. The coupled equations of motion are written in standard form as

$$M\ddot{q} + C\dot{q} + Kq = F \tag{15}$$

where M and K are the mass and stiffness matrices respectively, and C is the damping matrix calculated from assumed inherent damping ratios ζ_i. Using modal analysis the equations are transformed from a set of coupled equations in physical coordinates to a set of

coupled equations in physical coordinates to a set of uncoupled equations in modal coordinates

$$\ddot{X} + 2\zeta\omega\dot{X} + \Lambda X = B^{o}F \tag{16}$$

where

$$\Lambda = \mathrm{diag}[\omega_1^2, \omega_2^2, \ldots \omega_{12}^2]$$

the ω_i's are the undamped natural frequencies

$$\zeta\omega_n = \mathrm{diag}[\zeta_1\omega_1, \zeta_2\omega_2, \ldots, \zeta_{12}\omega_{12}]$$

$$B^{o} = [\phi_{5,1}, \phi_{5,2}, \ldots, \phi_{5,12}]^{T}$$

and the $\phi_{5,i}$'s are the fifth element of each eigenvector (the sensor and actuator are at the 5th DOF) normalized so

$$\Phi^{T}M\Phi = I$$

Equation 16 is rewritten in state space form as

$$\begin{bmatrix}\ddot{X}_p \\ \dot{X}_p\end{bmatrix} = \begin{bmatrix}-2\zeta_p\omega_{np} & -\Lambda_p \\ I & 0\end{bmatrix}\begin{bmatrix}\dot{X}_p \\ X_p\end{bmatrix} + \begin{bmatrix}B_p^{o} \\ 0\end{bmatrix}U_p \tag{17a}$$

$$Y_p = \begin{bmatrix}B_p^{oT} & 0\end{bmatrix}\begin{bmatrix}\dot{X}_p \\ X_p\end{bmatrix} \tag{17b}$$

where the subscript p denotes the equations apply to the plant. The reference model takes the same form,

$$\begin{bmatrix}\ddot{X}_m \\ \dot{X}_m\end{bmatrix} = \begin{bmatrix}-2\zeta_m\omega_{nm} & -\Lambda_m \\ I & 0\end{bmatrix}\begin{bmatrix}\dot{X}_m \\ X_m\end{bmatrix} + \begin{bmatrix}B_m^{o} \\ 0\end{bmatrix}U_m \tag{18a}$$

$$Y_m = \begin{bmatrix}B_m^{oT} & 0\end{bmatrix}\begin{bmatrix}\dot{X}_m \\ X_m\end{bmatrix} \tag{18b}$$

where the subscript m applies to the reference model.

For the purpose of numerical simulations we must reduce the size of the actual plant. In this example (as in ref. 2) we consider only

the first three modes of the actual plant and choose a reference model that includes only 2 modes. Damping ratios in the plant are assumed to be 0.01 while the desired damping ratios of the reference model are set at 0.05. All other parameters (length, EI, etc.) are set to 1.0 for convenience in calculations. In the present study we consider only initial condition responses. The first three modal states were initially set to 1.0, while all others were set to zero. From figures 3-4 it can be seen that the controller does an excellent job of forcing the actual plant to follow the reference model. However, from Table 1 it can be seen that the addition of the concentrated mass, i.e. errors between the actual plant and the reference model, can produce very large increases in the maximum control force and control effort needed for the controller to function. For example, a concentrated mass weighing 20% of the beam weight causes a factor of six increase in the quadratic control effort. This large increase in control effort demonstrates a need to find a method for choosing a good reference model.

SLEWING GRID EXAMPLE

The Virginia Tech slewing grid laboratory structure shown in figure 5 is a more complex example. The slewing grid was designed to have characteristics of LSS, namely closely spaced modes, low vibration frequencies, and low inherent damping. Three pairs of velocity sensors and force actuators are colocated at joints 3,4 &5. The slewing grid was designed to include a zero frequency rigid body rotation mode about the shaft, but this has never been realized because of bearing friction and slight misalignments of the rotational shaft. Although the geometry of the structure is symmetric about a horizontal line through joint 3, the vibration mode shapes are not similarly symmetric because the structure's weight causes asymmetric member gravity loading and therefore asymmetric stiffness distribution. It was considered desirable in the design phase to have at least one beam member in substantial compression relative to its buckling load, both to reduce the overall structural stiffness and to permit the possibility of nonlinear response. The lower horizontal member carries the largest compressive load, being compressed to about 70% of its Euler (pin-ended) buckling load. Great effort has been taken to accurately predict the loads in each member of the structure. However, each joint is held in place with a nut and bolt assembly and the process of tightening these bolts induces forces which we have been unable to determine accurately. Therefore our current Finite Element Model (FEM) only takes gravity forces into account. The rotational shaft was modeled by 8 beam finite elements with a displacement and rotation DOF at each node. Each of the 5 members of the structure is modeled with 4 finite elements which include a transverse displacement, an in-plane rotation, and an out of plane rotation at each node. The complete FEM has 72 DOF and the coupled equations of motions can be written

$$M\ddot{X} + C\dot{X} + (K + G)X = F \tag{19}$$

where G is the geometric stiffness matrix. To make the problem more manageable we created a reduced eleventh order model using the Guyan Reduction (ref. 15). The linear equations for the slewing grid can be written (in physical coordinates)

$$\begin{bmatrix} \ddot{X}_p \\ \dot{X}_p \end{bmatrix} = \begin{bmatrix} -M_p^{-1}C_p & -M_p^{-1}\bar{K_p} \\ I & 0 \end{bmatrix} \begin{bmatrix} \dot{X}_p \\ X_p \end{bmatrix} + \begin{bmatrix} B_p \\ 0 \end{bmatrix} U_p \tag{20a}$$

$$Y_p = \begin{bmatrix} B_p^T & 0 \end{bmatrix} \begin{bmatrix} \dot{X}_p \\ X_p \end{bmatrix} \tag{20b}$$

where

$$\bar{K_p} = K_p + G_p \tag{21}$$

M_p, C_p, K_p, and G_p are the reduced mass, damping, stiffness and geometric stiffness matrices, and B_p is (22 X 3) matrix with only 3 non-zero elements for mapping the control inputs to the proper DOF at joints 3,4,5.

The accuracy of the frequencies and modes predicted by the FEM is not good. During the past two years, great pains have been taken to find a FEM which would accurately model the structure. However all the non-linearities in the structure, such as friction in the bearing, large gravity loading in the lower horizontal member, and the loads induced by tightening the bolts at each joint of the structure, have resulted in a modeling nightmare. Table 2 compares frequencies predicted by our best FEM to the experimental vibration frequencies, and figures 6-9 compare several experimental and theoretical mode shapes. The difficulty in accurately modeling this structure was the driving force behind the decision to apply adaptive control to the slewing grid. The challenges of modeling the slewing grid may be similar to those we will face when we begin to model LSS.

In order to study the performance of MRAC for this case we first had to choose a model for simulating the actual plant. Our efforts to model the slewing grid as accurately as possible resulted in several FEM with varying degrees of accuracy. The most accurate model used experimental frequencies and mode shapes in a correction method proposed by Baruch (ref.16) to force the theoretical model to have exact experimental frequencies. The least accurate model was the standard FEM with no corrections. With this in mind we chose the most accurate FEM to simulate the plant and a linear combination of the

most accurate and least accurate model as the reference model. The reference model is described by the following equation.

$$Ref.Model = \alpha(standard\ FEM) + (1-\alpha)(corrected\ model) \qquad (22)$$

Thus we can vary the amount of error between the reference model and the actual model and monitor the control effort and system response as the error increases (α = 0---Perfect Modeling , α = 1--- Max. Error). Damping ratios for the simulated plant were obtained experimentally, while the damping ratios for the reference model specify the desired performance (see table 3).

In addition to varying the parameter α, we also varied the initial conditions of the structure. In the first simulation the structure was deformed into the second mode shape (see figure 6) and released. Table 4 shows that introducing errors into the reference model had a significant effect on the maximum force at joint 3 and the quadratic control effort. The maximum force required at joint 3 is 10 times larger at α = 1.0 than at α = 0.0, and the quadratic control effort is increased by a factor of 27 over the same interval. In the second simulation the structure was deformed into the theoretical fourth mode shape (see figure 8) and released. Table 5 shows that introducing errors in this case also causes dramatic increases in control costs. For example, at the point of maximum error (α = 1), the total control effort needed increased by almost 2500%, the maximum control force required by actuator 3 increased over 600%, and the amount of potential and kinetic energy in the system increased over 550%.

CONCLUDING REMARKS

The performance of Model Reference Adaptive Control (MRAC) was studied in numerical simulations with the objective of understanding the effects of differences between the plant and the reference model. MRAC was applied to two structural systems with controlled error between the reference model and the actual plant. Performance indices relating to control effort and response characteristics were monitored in order to determine what effects small errors have on the control effort and performance of the two systems. It was shown that reasonable amounts of error in the reference model can cause dramatic increases in both the control effort and and response magnitude (as measured by energy integrals) of the plant.

ACKNOWLEDGEMENT

This work was supported in part by NASA grant NAG-1-603

REFERENCES

1. Sobel, K., Kaufman, H., and Mabius, L., "Implicit Adaptive Control Systems for a Class of Multi-Input Multi-Output Systems," IEEE Transactions on Aerospace and Electronics Systems, Sept. 1982, pp. 576-590.

2. Bar-Kana, Kaufman, H., and Balas, M., "Model Reference Adaptive Control of Large Structural Systems," Journal of Guidance and Control 1983, Vol. 6, No. 2, pp.112-118.

3. Mufti, H., "Model Reference Adaptive Control for Large Structural Systems," Journal of Guidance and Control 1987, vol 10, No. 5, pp.507-509.

4. Ih, C-H.C., D. S. Bayard, and S. J. Wang, "Space Station Adaptive Payload Articulation Control," In Fourth IFAC Symposium on Control of Distributed Parameter Systems, Los Angles, CA, July 1986.

5. Bayard, D. S., C-H.C. Ih, and S. J. Wang, "Adaptive Control for Flexible Structures with Measurement Noise,: Proc. American Control Conference, Minneapolis, MN, pp.368-379, June 10-12 1987.

6. R. D. Nussbaum, "Some Remarks on a Conjecture in Parameter Adaptive Control," Systems and Control Letters, Vol. 3, No. 5 pp, 243-246, 1983.

7. J. C. Willems and C. I. Byrnes, "Global Adaptive Stabilization in the Absence of Information on the sign of the High Frequency Gain." Proc. INRIA Conf. on Analysis and Optimization of Systems, Springer-Verlag, 1984, pp.49-57.

8. C. I. Byrnes, and J. C. Willems, "Adaptive Stabilization of Multivariable Linear Systems,": IEEE Proc. 23 Conf. Decision and Control, 1984, pp. 1574-1577.

9. R. L. Kosut, "Adaptive Control of Large Space Structures," Proc. of the 5th AFOSR Forum on Space Structures, pp. 51-55, Aug. 20-21, 1987.

10. R. H. Middleton, G. C. Goodwin, D. J. Hill, and D. Q. Mayne, "Design Issues in Adaptive Control," IEEE Transactions on Automatic Control, Vol. 33, No. 1, pp.50-57, Jan, 1988.

11. N. Sundararajan, J. P. Williams, and R. C. Montgomery, "Adaptive Modal Control of Structural Dynamic Systems Using Recursive Lattice Filters," Journal of Guidance and Control, Vol. 8, No. 2, pp. 223-229, March-April, 1985.

12. C-H.C. Ih, D. S. Bayard, S. J. Wang, and D. B. Elder, "Adaptive Control Experiment With a Large Flexible Structure,:AIAA Guidance, Navigation and Control Conference, Paper AIAA-88-4153 -CP, Minneapolis, MN, Aug 15-17, 1988.

13. C-H.C. Ih, D. S. Bayard, A. Ahmed, and S. J. Wang, "Experiments in Multivariable Adaptive Control of a Large Flexible Structure," AIAA Guidance, Navigation and Control Conference, Boston, MA, Aug. 1989.

14. W-W. Chiang, and R. H. Cannon, Jr, "The Experimental Results of a Self Tuning Adaptive Controller Using Online Frequency Identification," The Journal of the Astronautical Sciences, Vol. 33, No. 1, Jan-Mar, 1985, pp. 71-83.

15. R. J. Guyan, "Reduction of Stiffness and Mass Matrices", AIAA Journal, Vol. 3, No 2, Feb. 1965.

16. M. Baruch, "Optimal Correction of Mass and Stiffness Matrices Using Measured Modes", AIAA Journal, Mol. 20, 1982, pp. 1623-1626.

TABLE 1

SUMMARY OF PERFORMANCE MEASURES FOR VARYING
AMOUNTS OF MASS ERROR IN THE SIMPLY SUPPORTED BEAM EXAMPLE

% ERROR	MAXIMUM FORCE	TOTAL QUADRATIC CONTROL EFFORT
0.0	380	11,422
10.0	540	27,486
20.0	920	72,263

TABLE 2

SUMMARY OF THEORETICAL VS EXPERIMENTAL FREQUENCIES
FOR THE SLEWING GRID STRUCTURE

MODE #	FREQUENCIES (Hz)		% ERROR
	THEORETICAL	EXPERIMENTAL	
1	0.36	0.42	16.67
2	1.37	1.45	5.84
3	3.00	2.88	4.00
4	4.47	5.39	20.58
5	6.02	6.41	6.48
6	6.69	6.88	2.84
7	9.79	9.05	7.56
8	11.52	10.18	11.63
9	13.11	13.56	3.43
10	15.35	14.90	2.93
11	21.16	15.37	27.36

TABLE 3

DAMPING RATIOS FOR THE MODELS OF THE SLEWING GRID

MODE #	EXPERIMENTAL FREQUENCY (HZ)	EXPERIMENTAL DAMPING RATIOS	DAMPING RATIOS FOR THE REF. MODEL
1	0.42	0.110	0.15
2	1.45	0.015	0.05
3	2.88	0.011	0.05
4	5.39	0.008	0.05
5	6.41	0.003	0.05
6	6.88	0.011	0.05
7	9.05	0.003	0.05
8	10.18	0.003	0.05
9	13.56	0.002	0.01
10	14.90	0.002	0.01
11	15.37	0.002	0.01

TABLE 4

SUMMARY OF PERFORMANCE MEASURES FOR VARYING AMOUNTS
OF ERROR IN THE SLEWING GRID
INITIAL CONDITIONS = MODE SHAPE 2

ALPHA	MAXIMUM FORCE (LBS)			QUADRATIC CONTROL EFFORT LBS^2-SEC	KINETIC ENERGY INTEGRAL LB-IN-SEC	POTENTIAL ENERGY INTEGRAL LB-IN-SEC
	JT. 3	JT. 4	JT. 5			
0.00	0.0182	0.022	0.021	0.00044	0.116	0.117
0.25	0.0596	0.028	0.025	0.00107	0.114	0.119
0.50	0.1090	0.038	0.035	0.00299	0.112	0.120
0.75	0.1520	0.048	0.045	0.00742	0.111	0.122
1.00	0.1825	0.057	0.058	0.01250	0.110	0.125

TABLE 5

SUMMARY OF PERFORMANCE MEASURES FOR VARYING AMOUNTS
OF ERROR IN THE SLEWING GRID
INITIAL CONDITIONS = MODE SHAPE 4

ALPHA	MAXIMUM FORCE (LBS)			QUADRATIC CONTROL EFFORT LBS^2-SEC	KINETIC ENERGY INTEGRAL LB-IN-SEC	POTENTIAL ENERGY INTEGRAL LB-IN-SEC
	JT. 3	JT. 4	JT. 5			
0.00	0.44	0.68	0.58	0.35	0.60	0.60
0.25	0.58	0.78	0.64	0.52	0.70	0.72
0.50	1.19	0.93	0.72	1.47	1.09	1.13
0.75	2.42	1.04	0.89	5.20	2.49	2.55
1.00	3.12	1.04	0.90	9.05	3.94	3.93

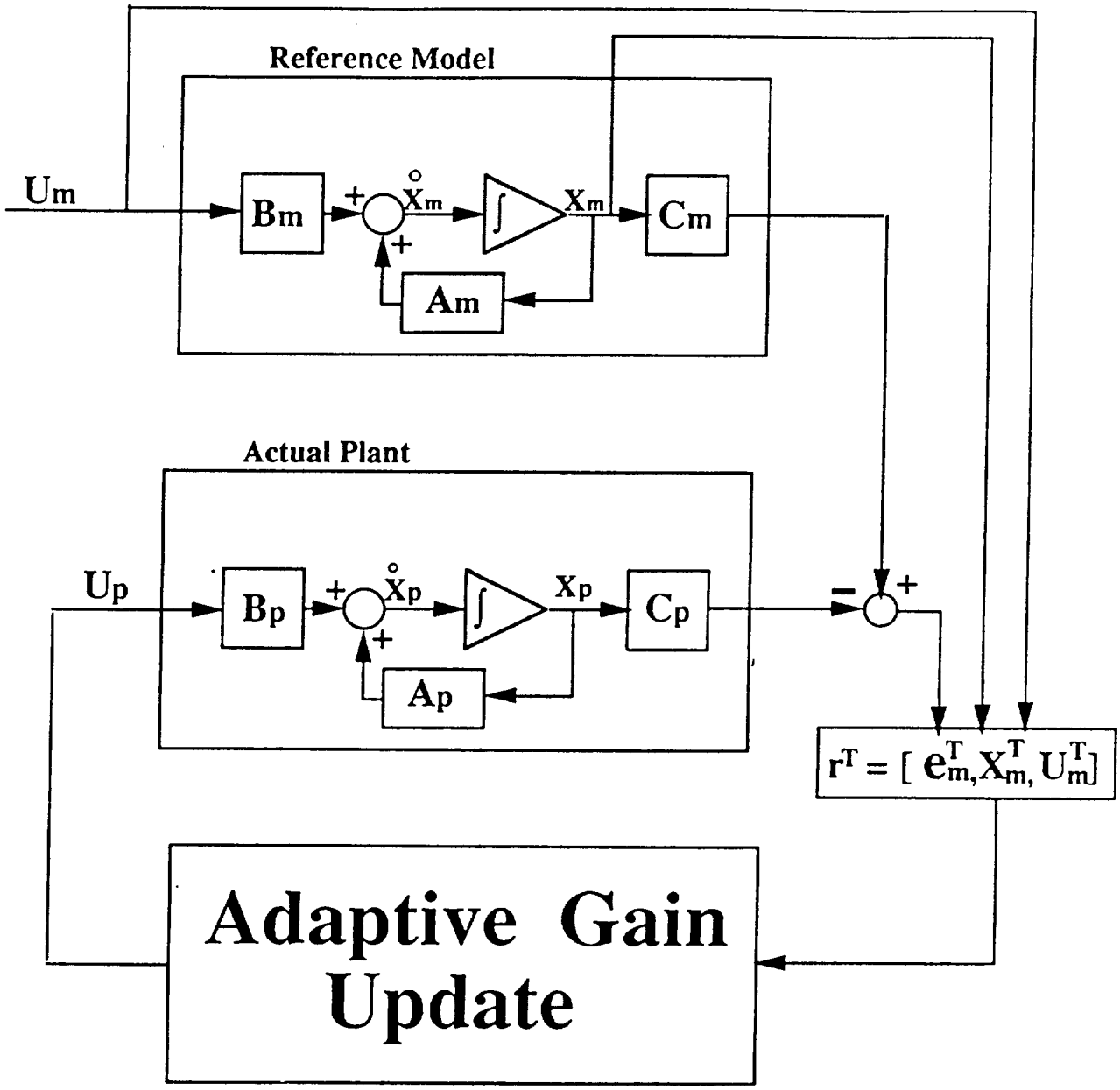


Figure 1

MODEL REFERENCE ADAPTIVE CONTROL BLOCK DIAGRAM

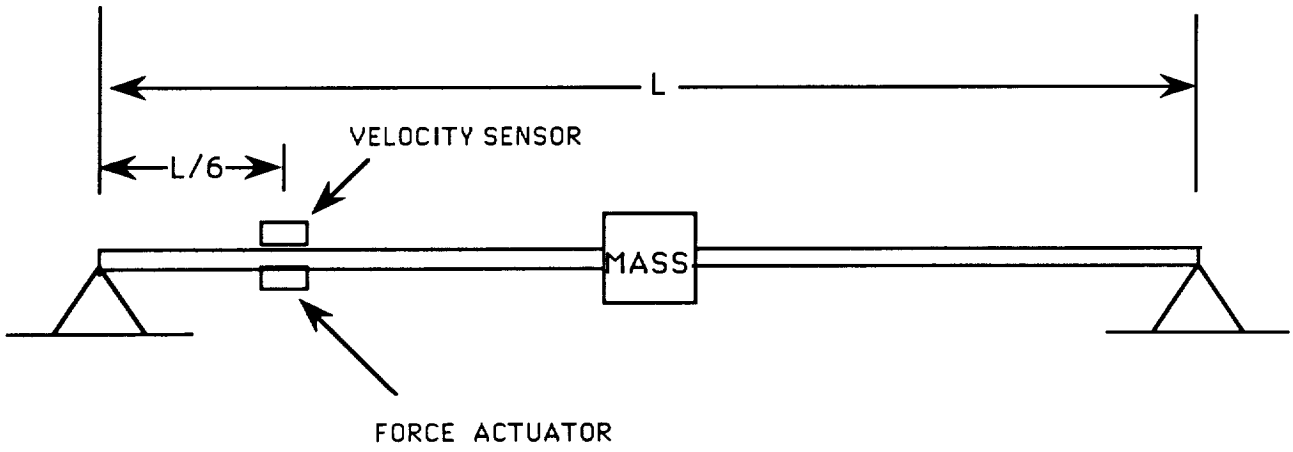


Figure 2

SIMPLY SUPPORTED BEAM WITH
VARIABLE CONCENTRATED MASS

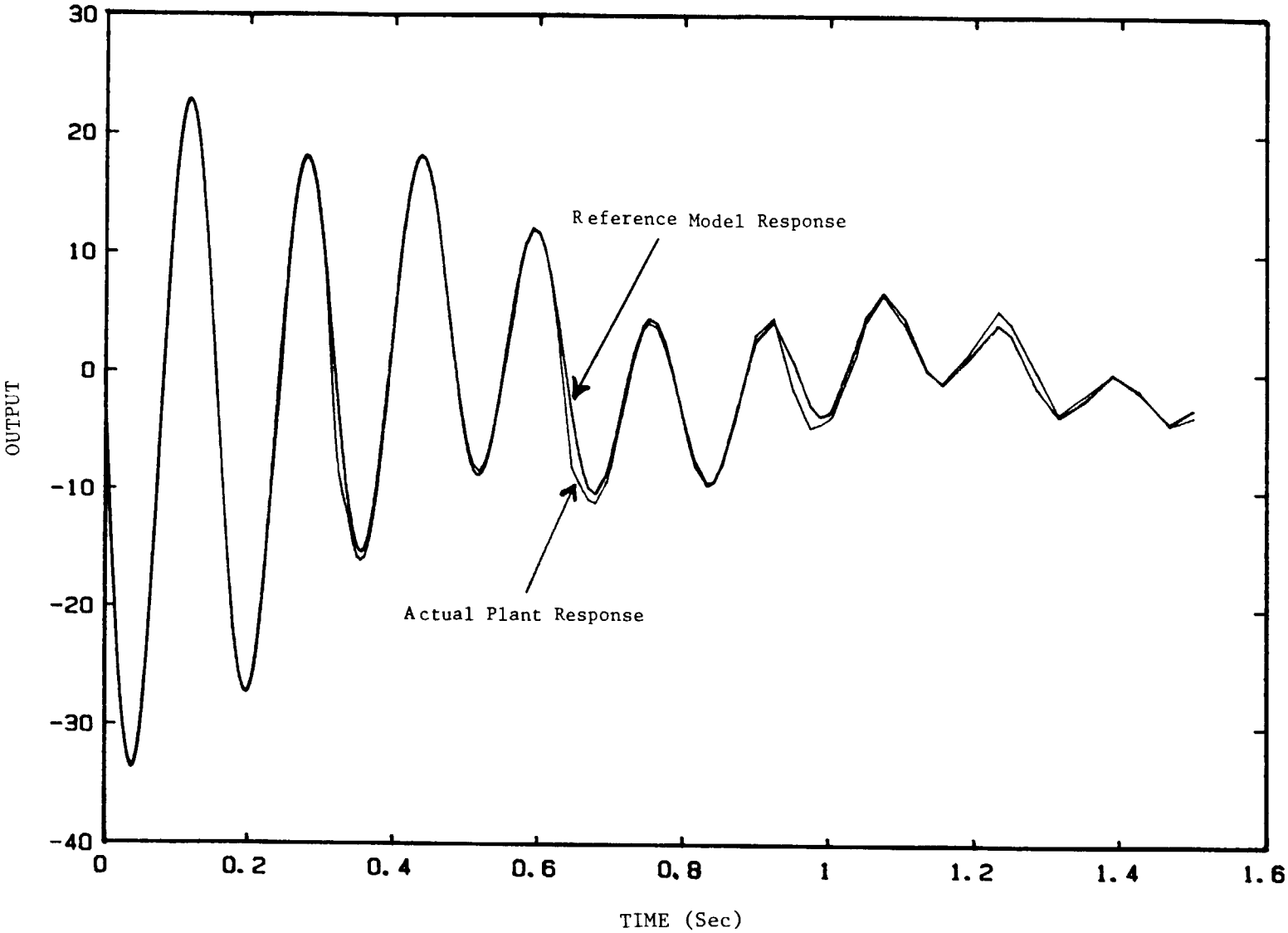


Figure 3
VELOCITY RESPONSE AT THE SENSOR LOCATION FOR THE
BEAM STRUCTURE WITH A 10% CONCENTRATED MASS

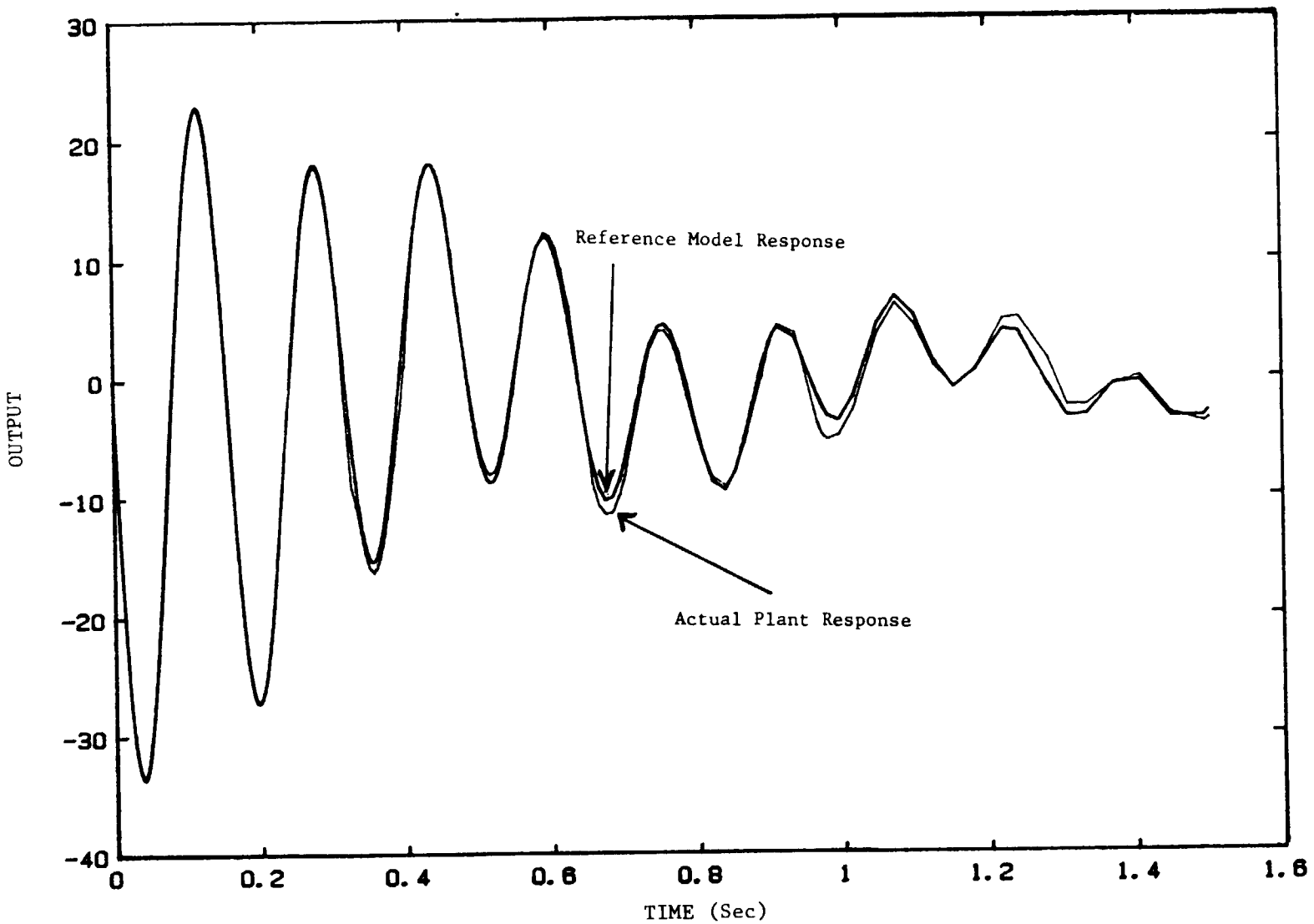
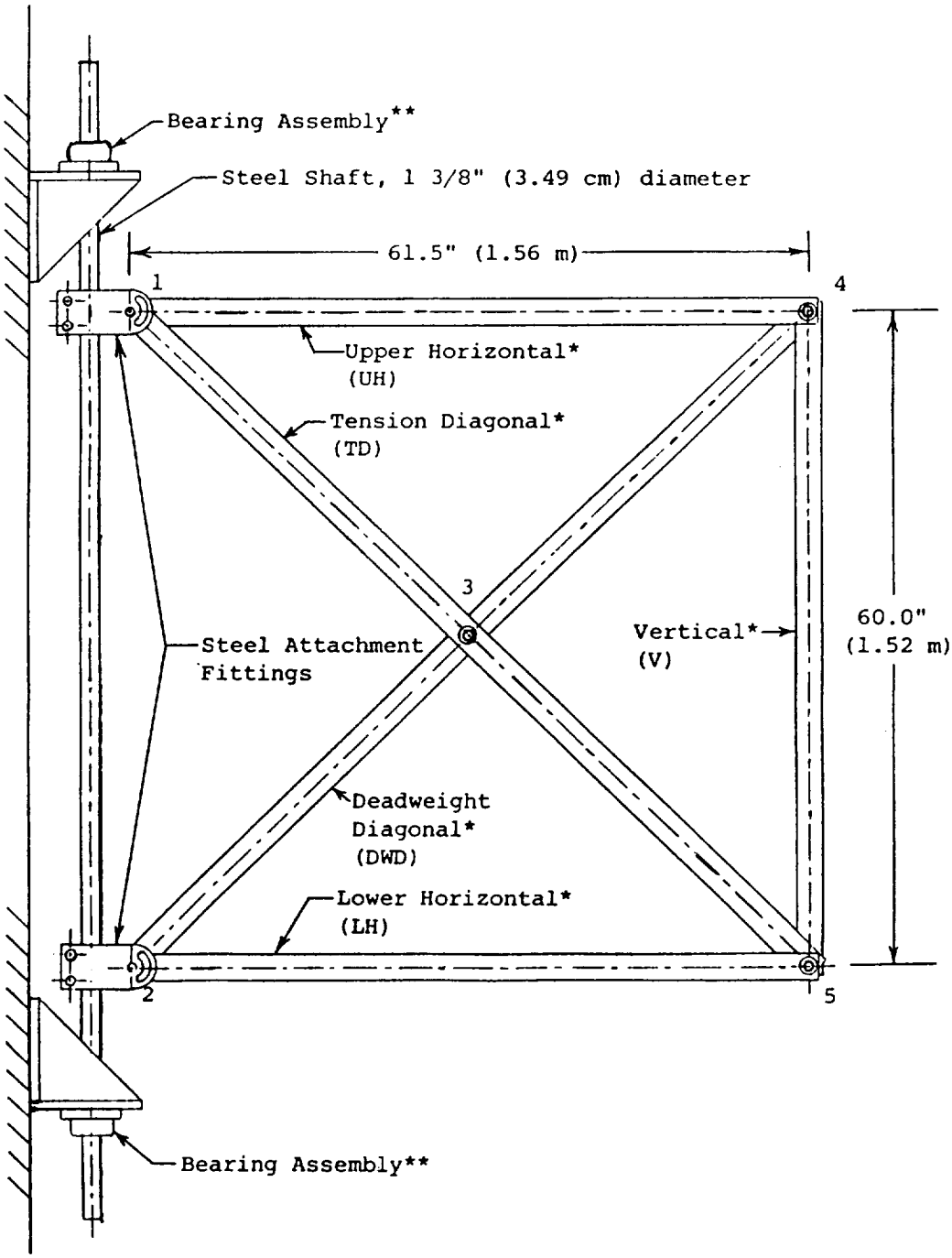


Figure 4
VELOCITY RESPONSE AT THE SENSOR LOCATION FOR THE
BEAM STRUCTURE WITH A 20% CONCENTRATED MASS



*Aluminum Beam Members
Alloy 6061-T6
Nominal cross-section:
2" x 1/8"
(5.08 cm x 0.32 cm)

**Ball Bearings
Make: SKF
Bearing No. 478207-106
Pillow Block Flange Unit
No. FYP-106
(Bearing seals and all grease were removed to reduce friction.)

Figure 5

SLEWING GRID STRUCTURE

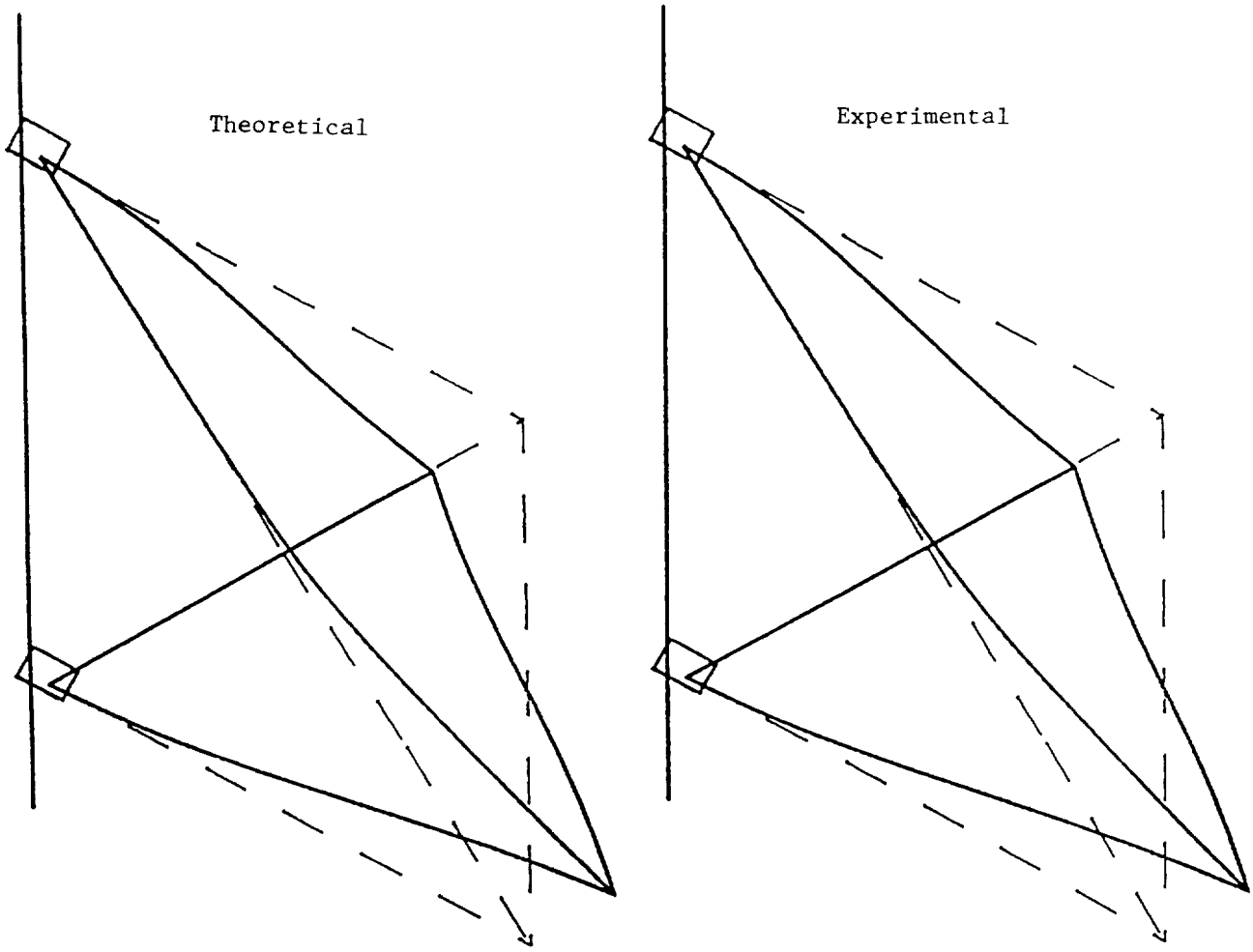


Figure 6

SECOND MODE SHAPE FOR SLEWING GRID STRUCTURE

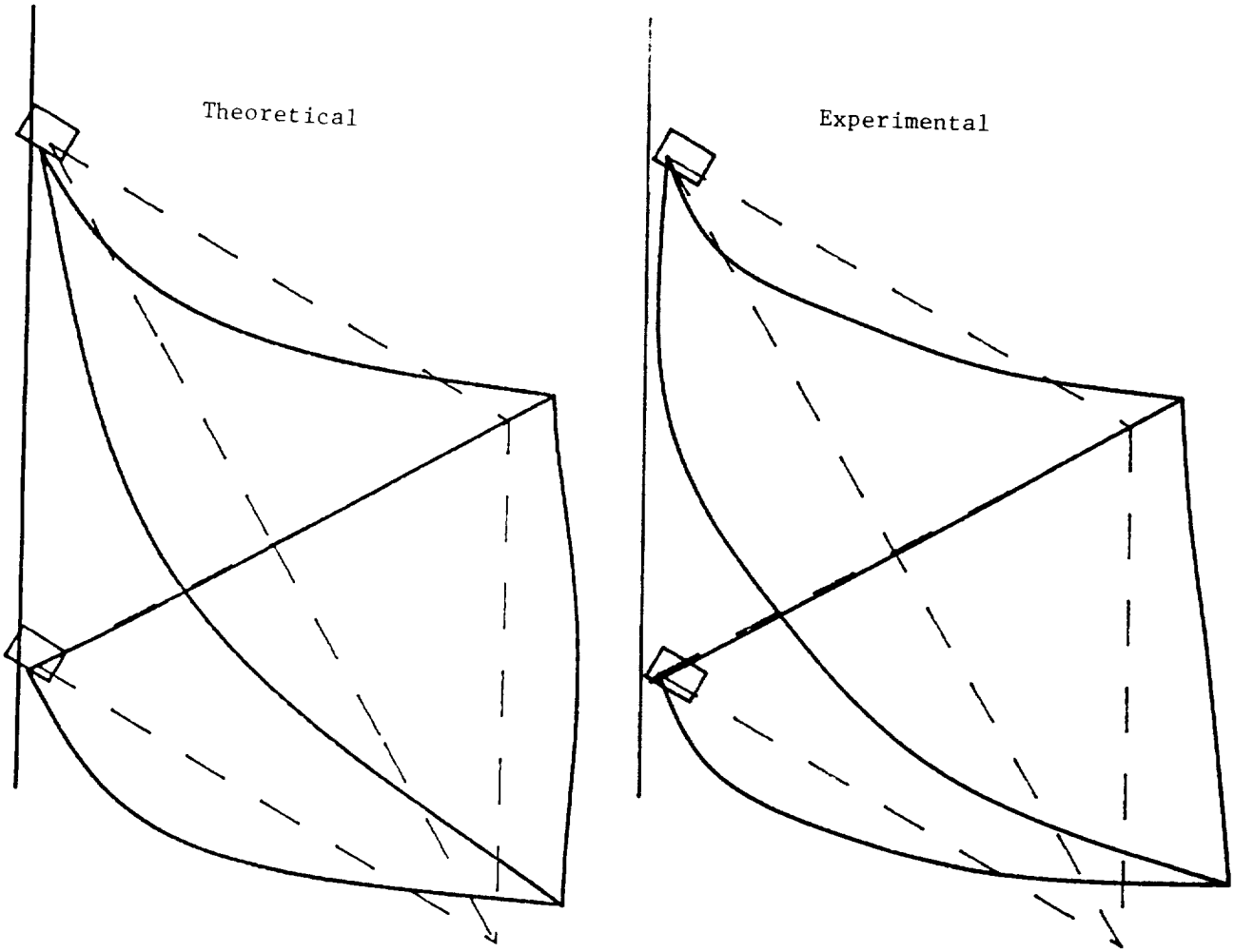


Figure 7

THIRD MODE SHAPE FOR SLEWING GRID STRUCTURE

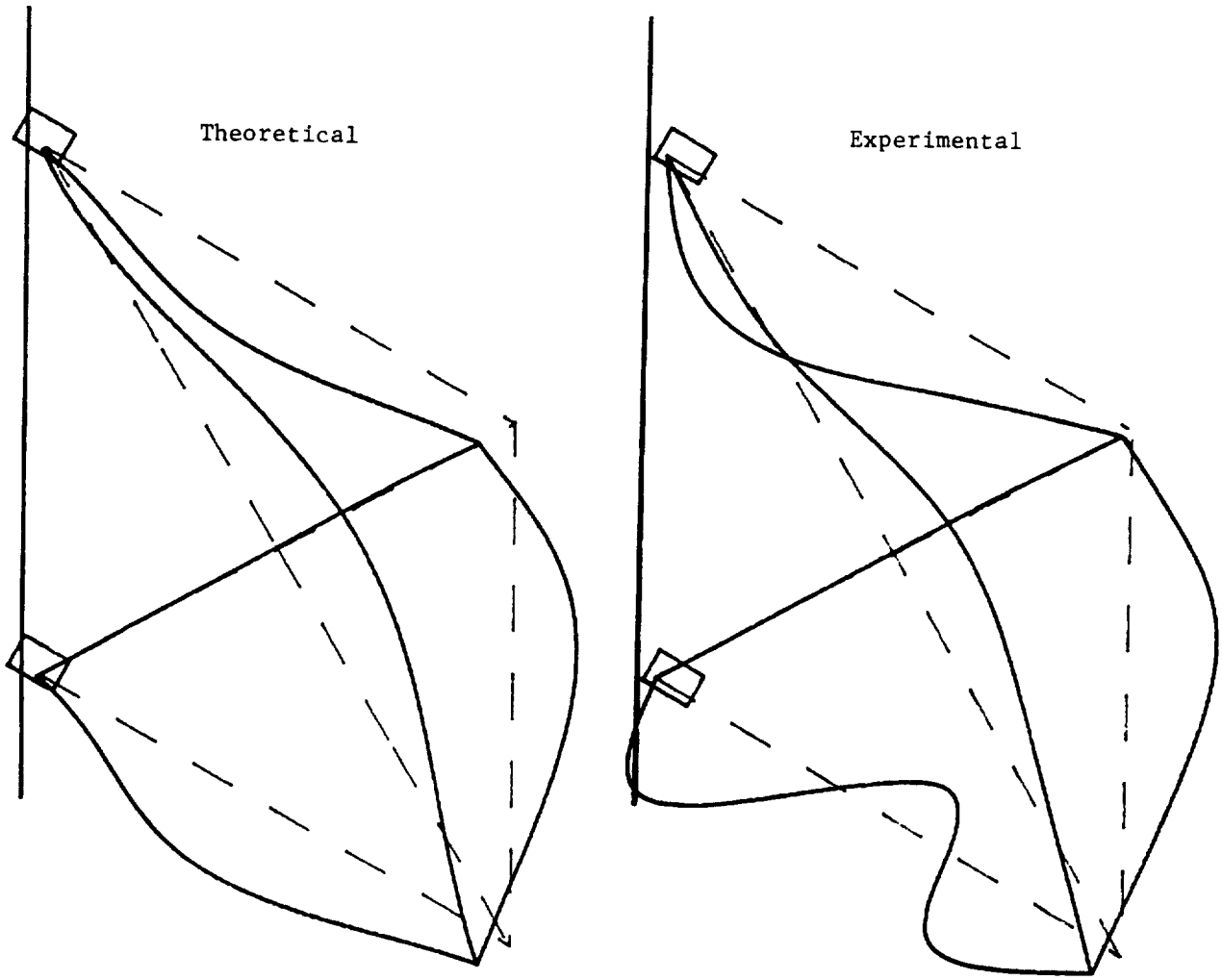


Figure 8

FOURTH MODE SHAPE FOR SLEWING GRID STRUCTURE

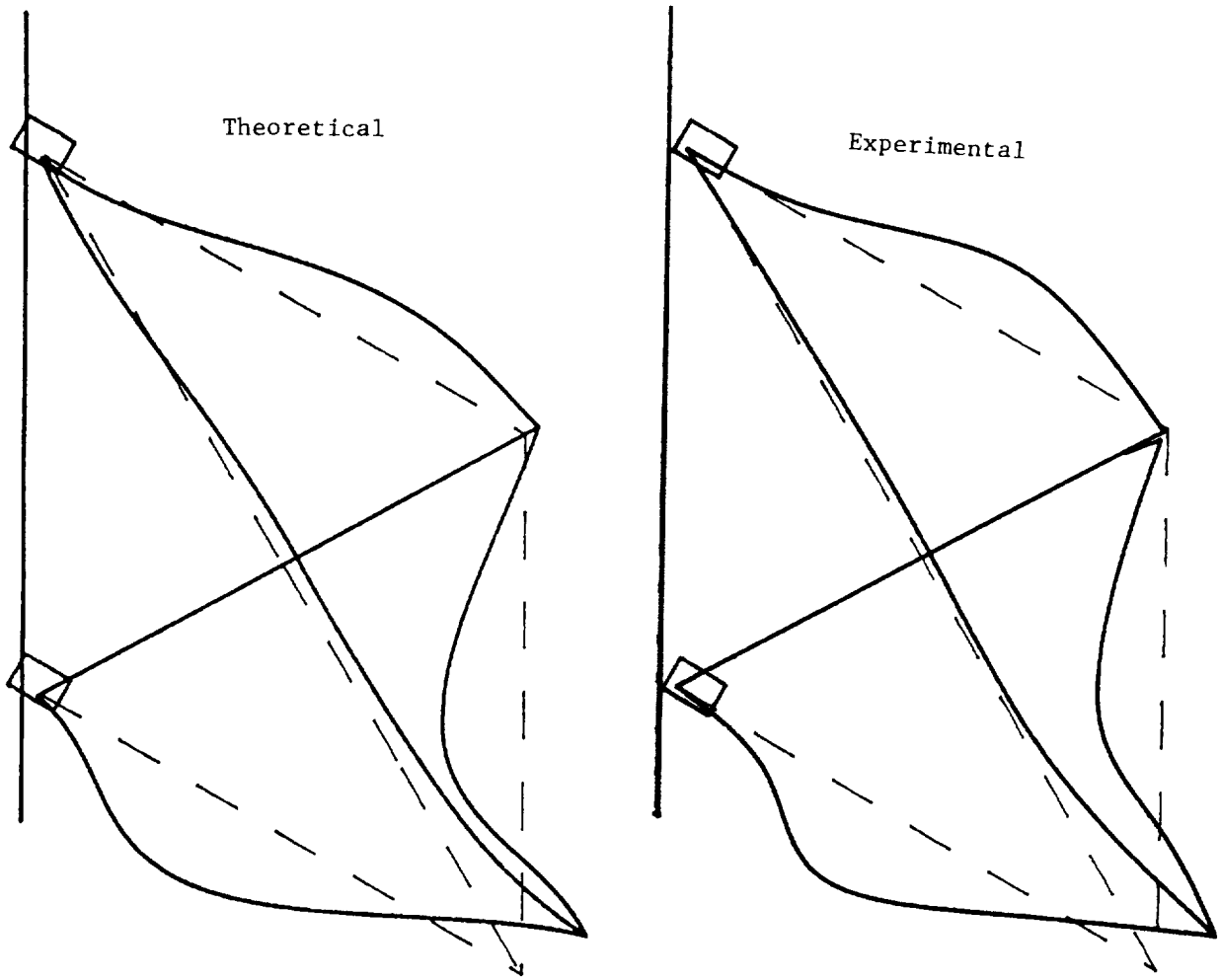


Figure 9

FIFTH MODE SHAPE FOR SLEWING GRID STRUCTURE

Component Mode Damping Assignment Techniques

Allan Y. Lee

Jet Propulsion Laboratory

July, 1990

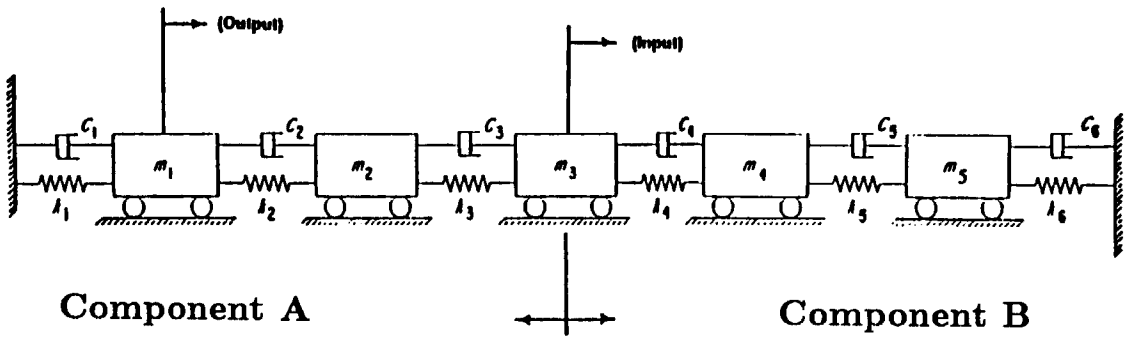
JPL

Background

- Multibody Dynamics Simulation Packages (e.g., DISCOS) Assemble Components' Dynamical Characteristics To Produce Dynamical Equations of A System of Interconnected Bodies (e.g., Galileo Spacecraft),
- For Flexible Components, Their Modal Characteristics Are Used As Inputs To The Package. In Particular, The Component Modal Damping Matrices Are Usually Assumed To be Diagonal,
- From Experience Obtained From Working With Large Space Structures, A Uniform Damping Factor of Not More Than 0.25% Can Usually Be Conservatively Assumed For All The System Modes,
- With Assumed Levels of Components' Damping, the Assembled System's Damping Might or Might Not Adhere to This "Rule of Thumb". If It Doesn't, A Time-consuming, Iterative Procedure Must Then Be Used to Adjust The Components' Damping Until It Does.

Objective

- **To Develop Techniques to Determine the Component Modes' Damping Factors Given the System Modes' Damping Factors**



System (5-DOF) = **Component A (3-DOF)** + **Component B (3-DOF)**

ω_k,	$k = 1,...,5$	ω_{Ai},	$i = 1,...,3$	ω_{Bj},	$j = 1,...,3$
ς_k,	$k = 1,...,5$	ς_{Ai},	$i = 1,...,3$	ς_{Bj},	$j = 1,...,3$

- **Given the System Modes' Damping Factors (ς_k, k = 1,...,5), Find the Component Modes' Damping Factors (ς_{Ai}, i = 1,...,3 and ς_{Bj}, j = 1,...,3).**

The Approach

- To Derive, from First Principles, a Relation Between the System Modal Damping Matrix and the Component Modal Damping Matrices,
- To Formulate and Solve an Optimization Problem that Enforce The Derived Component/System Modal Damping Relation.

Components' Equations of Motion in Physical and Modal Coordinates

$$[M_A][\ddot{x}_A] + [C_A][\dot{x}_A] + [K_A][x_A] = [G_A]u\,,$$

$$[x_A] = [V_A][q_A]\,,$$

$$[I_{n_A}][\ddot{q}_A] + [\bar{C}_A][\dot{q}_A] + [\bar{K}_A][q_A] = [\bar{G}_A]u\,.$$

A Similar Set of Equations May Be Written For Component B.

Remarks

- **Viscous Damping is a Simplified Mathematical Representation of a Rather Complex Situation Which Might Include Other Forms of Energy Dissipation (e.g., Hysteresis Damping, etc.).**
- **We Assume That If The Damping Is Small, These Damping Effects Can Be Grossly Represented By An Equivalent Viscous Term.**
- **$[\bar{C}_A]$ And $[\bar{C}_B]$ Are in General Non-diagonal, but Are Assumed to be Diagonal. Results Are Also Obtained Without This Assumption.**

System's Equations of Motion in Physical and Modal Coordinates

- **Physical Coordinates : Constructed from the Components' EOM Using the Compatibility Relations**

$$\begin{bmatrix} x_A \\ x_B \end{bmatrix} = \begin{bmatrix} [P_A] \\ [P_B] \end{bmatrix} [x] ,$$

$$[M][\ddot{x}] + [C][\dot{x}] + [K][x] = [G]u ,$$

Where $[M]$ May Be Expressed In Terms of $[M_A]$, $[M_B]$, $[P_A]$, and $[P_B]$. $[C]$ May Be Expressed In Terms of $[C_A]$, $[C_B]$, $[P_A]$, and $[P_B]$, etc.

- **Modal Coordinates**

$$[x] = [V][q] ,$$

$$[I_n][\ddot{q}] + [\bar{C}][\dot{q}] + [\bar{K}][q] = [\bar{G}]u ,$$

Where $[V]^T[M][V] = [I_n]$, $[\bar{G}] = [V]^T[G]$, $[\bar{K}] = [V]^T[K][V]$ (= Diag [ω_k^2], k = 1,...,5), And $[\bar{C}] = [V]^T[C][V]$ (Assumed to Be Diagonal).

One Way to Establish a Relation Between the System Generalized Coordinate $[q]$ and Those of the Components $[q_A]$ and $[q_B]$

Note that : $[x_A] = [V_A]\,[q_A] = [P_A]\,[x] = [P_A]\,[V]\,[q]$.

Hence

$$[q_A] = [Q_A][q]\,,$$

$$[q_B] = [Q_B][q]\,.$$

- $[Q_A](n_A \times n) = [V_A]^{-1}[P_A][V]$, $\qquad [Q_B](n_B \times n) = [V_B]^{-1}[P_B][V]$.
- $[V_A]$ and $[V_B]$ are non-singular matrices.
- **Relations Between Components' Modal Matrices and System's Modal Matrices**

$$[Q_A]^T[Q_A] + [Q_B]^T[Q_B] = [I_n]\,,$$

$$[Q_A]^T[\bar{C}_A][Q_A] + [Q_B]^T[\bar{C}_B][Q_B] = [\bar{C}]\,,$$

$$[Q_A]^T[\bar{K}_A][Q_A] + [Q_B]^T[\bar{K}_B][Q_B] = [\bar{K}]\,,$$

$$[Q_A]^T[\bar{G}_A] + [Q_B]^T[\bar{G}_B] = [\bar{G}]\,.$$

→ **Required Relation Between the System's Damping Matrix and the Components' Damping Matrices.**

Alternative Way To Express The Component/System Modal Damping Relation

$$\sum_{i=1}^{i=n_A} c_{Ai} R_{Ai} + \sum_{j=1}^{j=n_B} c_{Bj} R_{Bj} = [\bar{C}],$$

$$\sum_{i=1}^{i=n_A} \omega_{Ai}^2 R_{Ai} + \sum_{j=1}^{j=n_B} \omega_{Bj}^2 R_{Bj} = [\bar{K}].$$

Remarks :

- R_{Ai} **and** R_{Bj} **Are Determined From** Q_A **and** Q_B **Respectively,**
- R_{Ai} **and** R_{Bj} **May be Interpreted as "Weighting" Matrix that Determines the Contributions of the** i^{th} **Mode of Component A and** j^{th} **Mode of Component B to the System Damping,**
- **Same Weighting Matrices also Determine the Contributions of the Component Modes to the System Stiffness Matrix.**

Optimization Problem

$$\min_{c_{Ai},c_{Bj}} J = \frac{1}{2} \parallel \bar{C} - \sum_{i=1}^{i=n_A} c_{Ai} R_{Ai} - \sum_{j=1}^{j=n_B} c_{Bj} R_{Bj} \parallel_F^2 ,$$

where $\| \bullet \|_F^2$ is the Squared Frobenius Norm of the Matrix Concerned

The Optimality Conditions :

$$\frac{\partial J}{\partial c_{A1}} = \cdots = \frac{\partial J}{\partial c_{B1}} = \cdots = 0 , \qquad \begin{bmatrix} c_{A1} \\ c_{A2} \\ \cdots \\ c_{B1} \\ c_{B2} \\ \cdots \end{bmatrix} = A^{-1} \times B .$$

where A is a $(n_A + n_B) \times (n_A + n_B)$, and B is a $(n_A + n_B) \times 1$ matrices.

$$A = \begin{bmatrix} \| R_{A1}R_{A1} \| & .. & \| R_{A1}R_{An_A} \| & .. & \| R_{A1}R_{Bn_B} \| \\ \| R_{A2}R_{A1} \| & .. & \| R_{A2}R_{An_A} \| & .. & \| R_{A2}R_{Bn_B} \| \\ .. & .. & .. & .. & .. \\ \| R_{Bn_B}R_{A1} \| & .. & \| R_{Bn_B}R_{An_A} \| & .. & \| R_{Bn_B}R_{Bn_B} \| \end{bmatrix} , \qquad B = \begin{bmatrix} \| R_{A1}\bar{C} \| \\ \cdots \\ \| R_{An_A}\bar{C} \| \\ \| R_{B1}\bar{C} \| \\ \cdots \\ \| R_{Bn_B}\bar{C} \| \end{bmatrix} .$$

- $\| XY \| = \sum_{i=1}^{n} \sum_{j=1}^{n} X_{ij} \times Y_{ij}$
- From Cauchy's Inequality Theorem, the Determinant of A is Always Greater than Zero Unless the Matrices $R_{A1},\ldots,R_{An_A},\ldots,R_{Bn_B}$ are Linearly Dependent

Iterative Gradient Solution

$$[c_{A1}]_{k+1} = [c_{A1}]_k - \epsilon[\frac{\partial J}{\partial c_{A1}}]_k ,$$

$$\cdots$$

$$[c_{B1}]_{k+1} = [c_{B1}]_k - \epsilon[\frac{\partial J}{\partial c_{B1}}]_k ,$$

$$\cdots$$

- k = **Current Iteration Step**
- ϵ = **Small Positive Constant**

$$[\frac{\partial J}{\partial c_{A1}}]_k = -\sum_{r=1}^{r=n}\sum_{s=1}^{s=n}[R_{A1}]_{rs}([\bar{C}]_{rs} - \sum_{i=1}^{i=n_A}(c_{Ai})_k[R_{Ai}]_{rs} - \sum_{j=1}^{j=n_B}(c_{Bj})_k[R_{Bj}]_{rs}) ,$$

$$\cdots$$

$$[\frac{\partial J}{\partial c_{B1}}]_k = -\sum_{r=1}^{r=n}\sum_{s=1}^{s=n}[R_{B1}]_{rs}([\bar{C}]_{rs} - \sum_{i=1}^{i=n_A}(c_{Ai})_k[R_{Ai}]_{rs} - \sum_{j=1}^{j=n_B}(c_{Bj})_k[R_{Bj}]_{rs}) .$$

$$\cdots$$

- **The Iteration Stops When the Magnitudes of all the Gradients are Smaller Than a Prescribed Quantity (e.g., 10^{-10}).**

Modifications to the Optimization Problem

- **Weighting Matrix**

$$J_W = \frac{1}{2} \sum_{r=1}^{r=n} \sum_{s=1}^{s=n} W_{rs}^2 \alpha_{rs}^2$$

- **Inequality Constraints**

  - **Results Obtained From the Unconstrained Optimization Problem Might or Might not be "Physically Meaningful." Situations Arise in which, for a Given $[\bar{C}]$, One or More of c_{Ai} $(i = 1,...,n_A)$ and c_{Bj} $(j = 1,...,n_B)$ Might Be Negative**

  - **To Overcome the Difficulty, the Formulated Optimization Problem May be Modified with the Additions of Inequality Constraints**

$$c_{Ai} \geq 0, \qquad i = 1, ..., n_A,$$

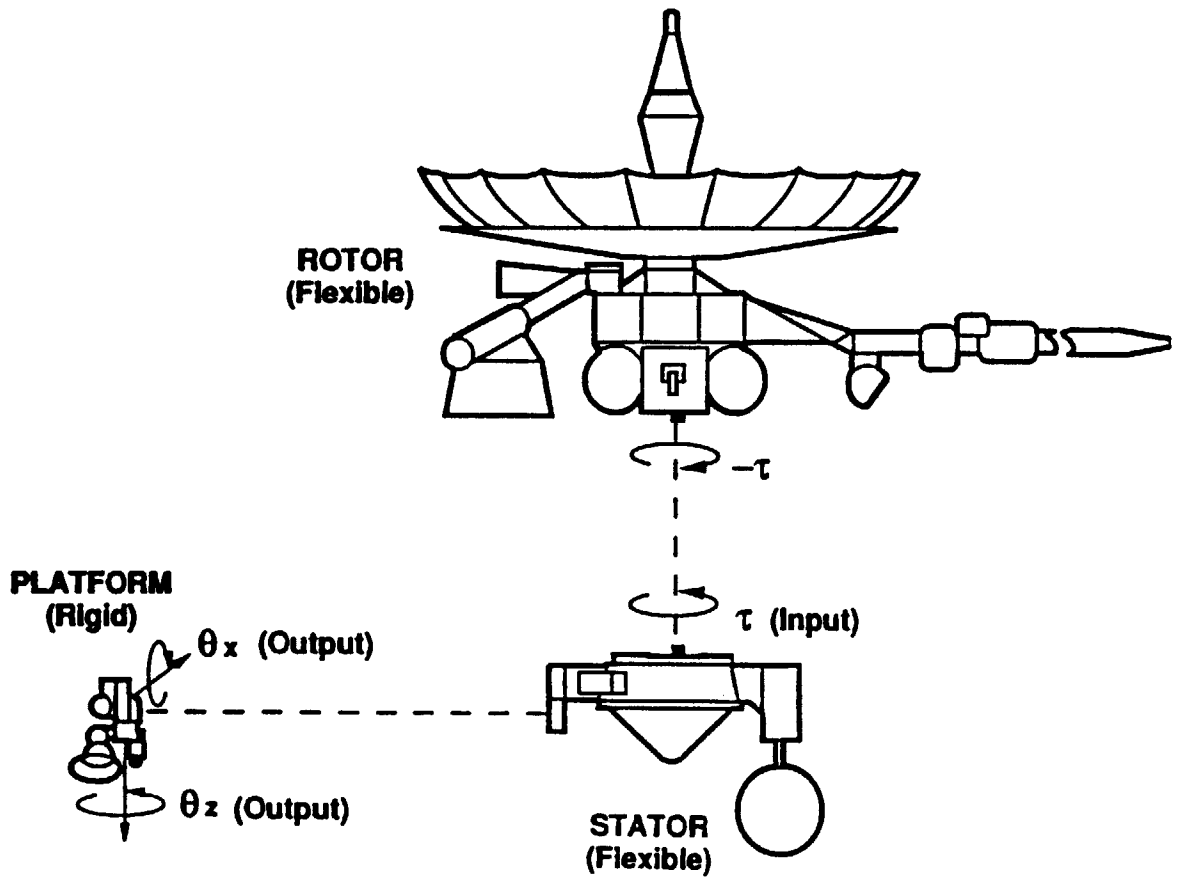
$$c_{Bj} \geq 0, \qquad j = 1, ..., n_B.$$

  - **No More Analytical (Algebric) Solution**

  - **May Be Solved Iteratively by, e.g., the Gradient Projection Method.**

Example : Galileo Spacecraft Cruise Model

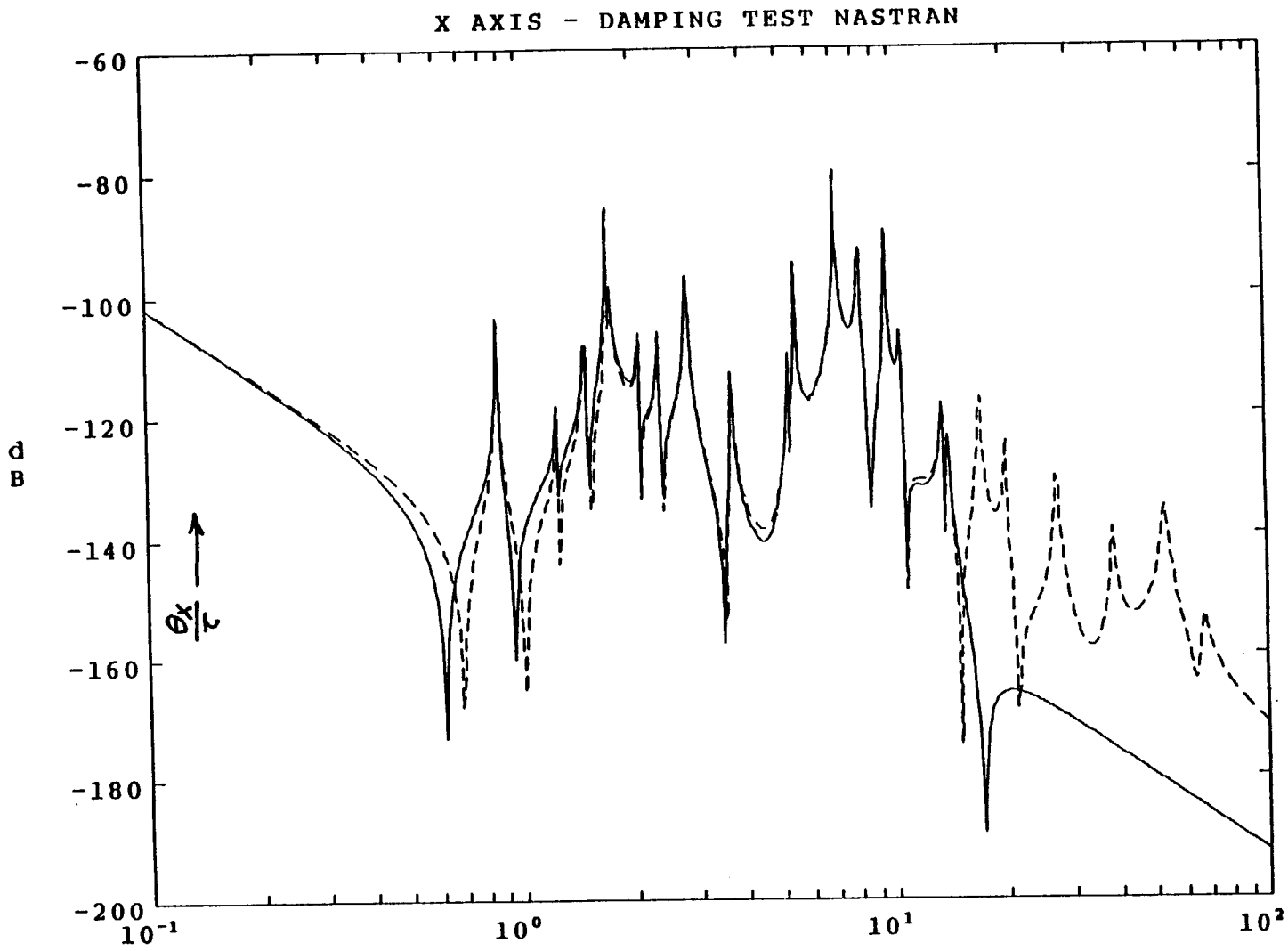
- A Complex High-order Model From Nastran FEM
- 26 Rotor Modes, 19 Stator Modes (After SVD), 6 Scan-platform Modes;
- Assembled System has 8 Rigid-body, 20 Retained, and 11 Extraneous Modes;
- Equal Weighting on All the Retained, and Zero Weighting on All Extraneous Modes.



Damping Ratios of the Reassembled System's Rigid-body, Retained, and Extraneous Modes

Mode	Frequency (Hz)	Rigid-body (%)	Retained (%)	Extraneous (%)
1	0	0		
2	0	0		
3	0	0		
4	0	0		
5	0	0		
6	0	0		
7	0	0		
8	0.0001	0		
9	0.8644		0.2500	
10	1.2355		0.2500	
11	1.4788		0.2500	
12	1.6782			0.5228
13	1.7070		0.2500	
14	1.7346		0.2500	
15	2.0724		0.2500	
16	2.3513		0.2500	
17	2.8152		0.2500	
18	3.7066		0.2500	
19	4.1724		0.2500	
20	5.2314		0.2500	
21	5.2473		0.2500	
22	5.4695		0.2500	
23	6.0124			22.4335
24	7.0593		0.2500	
25	7.1711			41.0165
26	8.1534		0.2500	
27	9.3513			14.5847
28	9.6102		0.2500	
29	10.3556			0.2146
30	10.4210		0.2500	
31	10.5549		0.2500	
32	13.5342		0.2500	
33	13.9894		0.2500	
34	17.0659			0.5842
35	19.9321			0.4905
36	27.2954			0.7231
37	38.4226			1.0406
38	53.0927			1.4643
39	66.6673			1.9455

Bode Plot, Galileo Spacecraft Cruise Model, (X-axis)



—— **Full-order Model**

– – – **Reduced-order Model**

Summary

(1) A Relation Between the System Modal Damping Matrix and the Component Modal Damping Matrices is Derived from First Principles;

(2) An Optimization Problem is then Formulated to Select All the Component Modes' Damping Ratios that Best Satisfy the Above Derived Relation;

(3) A Weighting Matrix is Used in the Cost Functional to Stress the Relative Importance of the Diagonal Terms in the Damping Matrix. Inequality Constraints are Also Added to the Optimization Problem to Pick Only Nonnegative Component Modes' Damping Factors;

(4) The Optimization Problem May be Solved Algebraically or Iteratively;

(5) The Proposed Techniques are Successfully Used on a High-order, Finite-element Model of the Galileo Spacecraft.

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CP-10065, Part 1		

4. Title and Subtitle	5. Report Date
4th NASA Workshop on Computational Control Of Flexible Aerospace Systems	March 1991
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Lawrence W. Taylor, Jr. (Compiler)	
	10. Work Unit No. 506-46-11-01
9. Performing Organization Name and Address	11. Contract or Grant No.
NASA Langley Research Center Hampton, VA 23665-5225	
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Conference Publication
National Aeronautics and Space Administration Washington, DC 20546	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

This publication is a collection of papers presented at the Fourth NASA Workshop on Computational Control of Flexible Aerospace Systems. The Workshop was held at the Kingsmill Resort, Williamsburg , Virginia, July 11-13, 1990. The papers address modeling, systems identification, and control of flexible aircraft, spacecraft and robotic systems.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Large Flexible Spacecraft Control Structural Dynamics	Unclassified - Unlimited Subject Category - 18

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	467	A20

NASA FORM 1626 OCT 86 For sale by the National Technical Information Service, Springfield, Virginia 22161-2171

NASA-Langley, 1991