NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Technical Report 32-1598

Attitude Dynamics Simulation Subroutines for Systems of Hinge-Connected Rigid Bodies With Nonrigid Appendages

G. E. Fleischer

P. W. Likins

(NASA-CR-145794) ATTITUDE DYNAMICS SIMULATION SUBROUTINES FOR SYSTEMS OF HINGE-CONNECTED RIGID BODIES WITH NONRIGID APPENDAGES (Jet Propulsion Lab.) 135 p HC $6.00 CSCL 22C G3/13

N76-12091 THRU N76-12095 Unclas 03846

JET PROPULSION LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
PASADENA, CALIFORNIA

August 15, 1975

Preface

The work described in this report was performed by the Guidance and Control Division of the Jet Propulsion Laboratory.

Contents

Figures

PRECEDING PAGE BLANK NOT FILMED

Abstract

This report describes three computer subroutines designed to solve the vector-dyadic differential equations of rotational motion for systems that may be idealized as a collection of hinge-connected rigid bodies assembled in a tree topology, with an optional flexible appendage attached to each body. Deformations of the appendages are mathematically represented by modal coordinates and are assumed small. Within these constraints, the subroutines provide equation solutions for (1) the most general case of unrestricted hinge rotations, with appendage base bodies nominally rotating at a constant speed, (2) the case of unrestricted hinge rotations between rigid bodies, with the restriction that those rigid bodies carrying appendages are nominally nonspinning, and (3) the case of small hinge rotations and nominally nonrotating appendages, i.e., the linearized version of case 2. Sample problems and their solutions are presented to illustrate the utility of the computer programs. Complete listings and user instructions are included for these routines (written in Fortran), which are intended as general-purpose tools in the analysis and simulation of spacecraft and other complex electromechanical systems.

Attitude Dynamics Simulation Subroutines for Systems of Hinge-Connected Rigid Bodies With Nonrigid Appendages

I. Introduction

Equations of motion which characterize the small, time-varying deformations of an elastic appendage attached to a rigid body experiencing arbitrary motions have been derived in detail for distributed-mass finite element models in Ref. 1, and for discrete mass models in Ref. 2. With the general structure of the appendage deformation equations established in these references, coordinate transformations are developed in Refs. 1 and 3 in order to allow representation of the elastic appendage in terms of a set of truncated modal coordinates far fewer in number than the original set. In Ref. 4, additional equations of motion are derived to describe the rotations of typical bodies in a *system* of hinge-connected rigid bodies arranged as a topological tree, with optional arbitrary *nonrigid* appendages attached to each rigid body in the system. In this respect, the results of Hooker in Ref. 5 and those of Ref. 4 are parallel.

It is the purpose of this report first to draw together the appendage equations and the equations describing rigid body motions of the tree system, assuming that some or all of the rigid bodies carry nonrigid appendages, and to derive a consistent and detailed set of system dynamical equations suitable for digital computer solution. Secondly, it is the purpose here to present general-purpose computer subroutines capable of solving the resulting system equations of rotational motion, and to demonstrate their utility and applicability to a wide class of spacecraft.

In generating the equations of motion for the hinge-connected tree of rigid bodies with nonrigid appendages, two specific formulations are obtained. The first formally constrains[1] appendage base motion to small deviations from a nominal constant angular velocity in inertial space, thus allowing appendage rotation but with only small deviations from a constant rate of spin. The second formulation formally permits no spin and constrains appendage base motion to small deviations from a nominally zero angular velocity (and acceleration) in the inertial frame. However, both formulations permit otherwise unrestricted motions of the system rigid bodies consistent with the fundamental assumption of small appendage deformations from some nominal state. Computer subroutines (written in Fortran) are described which solve the equations produced by each of these approaches. In addition, a third subroutine is presented which solves the completely linearized equations for the nonrotating case, under the assumption that all rigid body rotations and their derivatives are small.

The computer programs are direct descendants of those described in Refs. 6 and 7, which are applicable to the hinge-connected rigid body tree *without* nonrigid appendages. All of the programs are designed to calculate the angular accelerations for every rigid and nonrigid body in the system but do *not* perform numerical integration. Thus, the routines are intended as general-purpose tools, to be called into action by the user's own particular simulation language, whether this be CSSL, CSMP, MIMIC, or some "homemade" variety. Each of the routines allows the user to prescribe the motion of any *rigid* body in the system rather than allow it to be calculated, a feature often useful for eliminating unwanted dynamics or for "rigidizing" certain joints in sensitivity studies.

II. Unrestricted Systems

A. Mathematical Model

Any problem of dynamic analysis must begin with the adoption of a mathematical model representing the physical system of interest. In what follows, it is assumed that the model consists of $n + 1$ rigid bodies (labeled $\mathscr{b}_0, \ldots, \mathscr{b}_n$) interconnected by n line hinges (implying no closed loops and, hence, tree topology), with each body containing no more than three orthogonal rigid rotors, each with an axis of symmetry fixed in the housing body, and moreover with the possibility of attaching to each of the $n + 1$ bodies a nonrigid appendage, with appendage $\mathscr{a}_k$ attached to body $\mathscr{b}_k$.

If the actual connection between two massive portions of the physical system admits two (or three) degrees of freedom in rotation, then the analyst simply introduces one (or two) massless and dimensionless imaginary bodies into his model (as though they were massless gimbals). Since the number of equations to be derived here matches the number of degrees of freedom of the system, no price is paid in problem dimension by the introduction of imaginary bodies.

Each combination of a rigid body and its internal rotors and attached flexible appendage comprises a basic building block, referred to here as a *substructure*;

---

[1] Deviations from nominal appendage base motion are treated as small in the sense that their products with appendage deformations are ignored, but nonlinear terms in these base motion deviations alone are retained. Thus, there is a *formal* limitation to small base motion deviations from nominal, but in practical applications, substantial deviations are accommodated quite satisfactorily.

thus, there are $n + 1$ substructures in the total system, so labeled that s_k encompasses b_k, a_k, and any rotors in b_k.

Definitions and Notations

Definitions and notational conventions are as follows (see Fig. 1):

Def. 1. Let n be the number of hinges interconnecting a set of $n + 1$ substructures.

Def. 2. Define the integer set $\mathfrak{B} \equiv \{0, 1, \ldots, n\}$.

Def. 3. Define the integer set $\mathcal{P} \equiv \{1, \ldots, n\}$.

Def. 4. Let b_0 be a label assigned to one rigid body chosen arbitrarily as a reference body, and let $b_1, \ldots, b_n$ be labels assigned to the rest of the rigid bodies in such a way that if b_j is located between b_0 and b_k, then $0 < j < k$.

Def. 5. Define dextral, orthogonal sets of unit vectors $\mathbf{b}_1^k, \mathbf{b}_2^k, \mathbf{b}_3^k$ so as to be imbedded in b_k for $k \in \mathfrak{B}$, and such that in some arbitrarily selected nominal configuration of the total system, $\mathbf{b}_\alpha^k = \mathbf{b}_\alpha^j$ for $\alpha = 1, 2, 3$ and $k, j \in \mathfrak{B}$.

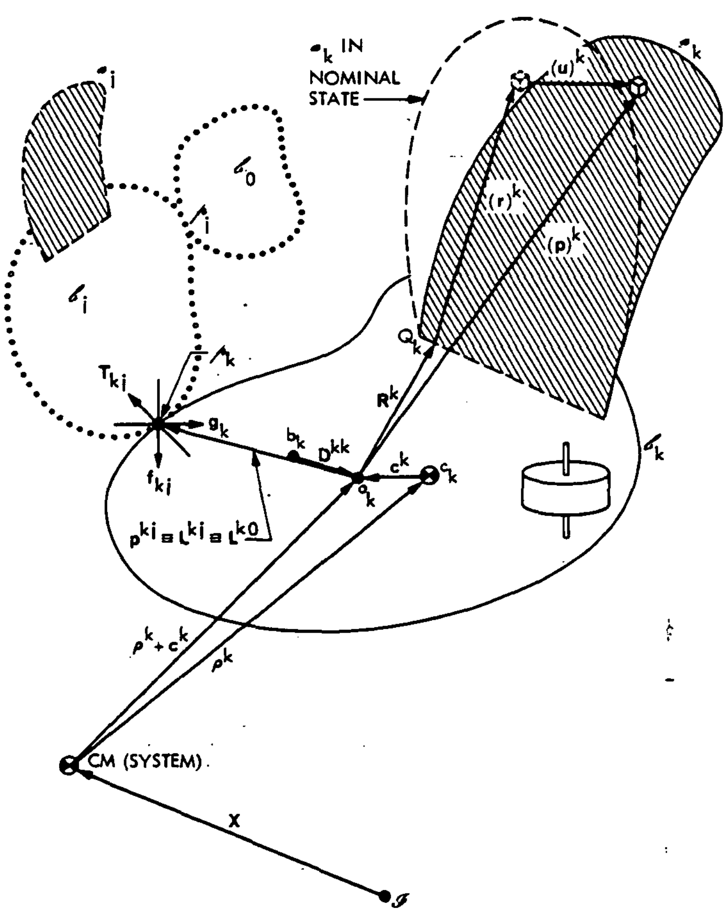

Fig. 1. Definitions for the *k*th substructure, with $j < k$

Def. 6. Define

$$\{\mathbf{b}^k\} \equiv \begin{Bmatrix} \mathbf{b}_1^k \\ \mathbf{b}_2^k \\ \mathbf{b}_3^k \end{Bmatrix} \quad k \in \mathscr{B}$$

Def. 7. Define $\{\mathbf{i}\}$ as a column array of inertially fixed, dextral, orthogonal unit vectors $\mathbf{i}_1$, $\mathbf{i}_2$, $\mathbf{i}_3$.

Def. 8. Let C be the direction cosine matrix defined by

$$\{\mathbf{b}^0\} = C\{\mathbf{i}\}$$

Def. 9. Let $\boldsymbol{\omega}^0 \equiv \{\mathbf{b}^0\}^T \omega^0$ be the inertial angular velocity vector of $\mathscr{b}_0$, so that ω^0 is the corresponding 3×1 matrix in basis $\{\mathbf{b}^0\}$.

Def. 10. Let c_k be the mass center of the kth substructure, $k \in \mathscr{B}$.

Def. 11. Let $\mathscr{h}_k$ be a point on the hinge axis common to $\mathscr{b}_k$ and $\mathscr{b}_j$ for $j < k$ and $k \in \mathscr{P}$.

Def. 12. Let $\mathbf{p}^{kj}$ be the position vector of the hinge point connecting $\mathscr{b}_j$ and $\mathscr{b}_k$ from the point o_k occupied by c_k when the kth substructure is in its nominal state.

Def. 13. Let $\mathbf{c}^k$ be the position vector from c_k to o_k.

Def. 14. Let $\boldsymbol{\rho}^k$ be the position vector to c_k from the system mass center CM.

Def. 15. Let $\mathbf{X}$ be the position vector to CM from an inertially fixed point $\mathscr{I}$, and let $X = \mathbf{X} \cdot \{\mathbf{i}\}$.

Def. 16. Let $\mathfrak{M}_k$ be the mass of the kth substructure, for $k \in \mathscr{B}$.

Def. 17. Let $(\mathbf{p})^k$ be a generic position vector from o_k to any point in the kth substructure.

Def. 18. Let Q_k be a point common to rigid body $\mathscr{b}_k$ and flexible appendage $\mathscr{a}_k$.

Def. 19. Let $\mathbf{R}^k = \{\mathbf{b}^k\}^T R^k$ be the position vector fixed in $\mathscr{b}_k$ locating Q_k with respect to o_k.

Def. 20. Let $(\mathbf{r})^k = \{\mathbf{b}^k\}^T (r)^k$ be a generic symbol such that $\mathbf{R}^k + (\mathbf{r})^k$ locates a typical field point in $\mathscr{a}_k$ with respect to o_k when the flexible appendage is in some nominal state (perhaps undeformed). For a discretized appendage $\mathscr{a}_k$, let $(\mathbf{r}^s)^k = \{\mathbf{b}^k\}^T (r^s)^k$ locate the sth node in the nominal state.

Def. 21. Define the generic deformation vector $(\mathbf{u})^k$ in such a way that[2]

$$(\mathbf{p})^k \equiv \mathbf{R}^k + (\mathbf{r})^k + (\mathbf{u})^k$$

and

$$(p)^k = R^k + (r)^k + (u)^k$$

[2] Superscripts on generic symbols such as **p**, **r**, and **u** will be omitted when obvious, as when the symbol appears within an integrand of a definite integral.

For a discretized appendage $\mathscr{a}_k$, let $(\mathbf{u}^s)^k = \{\mathbf{b}^k\}^T (u^s)^k$ be the deformation vector for node s.

Def. 22. Let $\mathbf{g}^k \equiv \{\mathbf{b}^k\}^T g^k$ be a unit vector parallel to the hinge axis through $\mathscr{h}_k$.

Def. 23. For $k \in \mathcal{P}$, let γ_k be the angle of a $\mathbf{g}^k$ rotation of $\mathscr{b}_k$ with respect to the body attached at $\mathscr{h}_k$. Let γ_k be zero when $\mathbf{b}_\alpha^k = \mathbf{b}_\alpha^j (\alpha = 1, 2, 3; j, k \in \mathcal{B})$.

Def. 24. Let $\mathbf{J}^k \equiv \{\mathbf{b}^k\}^T J^k \{\mathbf{b}^k\}$ be the inertia dyadic of the kth substructure for o_k, so that $\mathbf{J}^k$ is time-variable by virtue of deformations.

Def. 25. Let $\mathbf{F}^k \equiv \{\mathbf{b}^k\}^T F^k$ be the resultant vector of all forces applied to the kth substructure except for those due to interbody forces transmitted at hinge connections.

Def. 26. Let $\mathbf{T}^k \equiv \{\mathbf{b}^k\}^T T^k$ be the resultant moment vector with respect to c_k of all forces applied to the kth substructure except for those due to interbody forces transmitted at hinge connections.

Def. 27. Let τ_k be the scalar magnitude of the torque component applied to $\mathscr{b}_k$ in the direction of $\mathbf{g}^k$ by the body attached at $\mathscr{h}_k$.

Def. 28. Let $\mathbf{F} \equiv \sum_{k \in \mathcal{B}} \mathbf{F}^k = \{\mathbf{b}^0\}^T F$ be the external force resultant for the total system.

Def. 29. Define the scalar ϵ_{sk} such that for $k \in \mathcal{B}$ and $s \in \mathcal{P}$

$$\epsilon_{sk} \equiv \begin{cases} 1 & \text{if } \mathscr{h}_s \text{ lies between } \mathscr{b}_0 \text{ and } \mathscr{b}_k \\ 0 & \text{otherwise} \end{cases}$$

(The $n(n+1)$ scalars ϵ_{sk} are called *path elements.*)

Def. 30. Define $\mathcal{M} \equiv \sum_{k \in \mathcal{B}} \mathcal{M}_k$, the total system mass.

Def. 31. Let C^{rj} be the direction cosine matrix defined by $\{\mathbf{b}^r\} = C^{rj}\{\mathbf{b}^j\}$, $r, j \in \mathcal{B}$. (Note that in the nominal state, $C^{rj} = U$, the unit matrix.)

Def. 32. Let N_{kr} denote the index of the body attached to $\mathscr{b}_k$ and on the path leading to $\mathscr{b}_r$, and let $N_{kk} \equiv k$. (These are the *network elements.*) For notational simplicity, use N_k for N_{k0}.

Def. 33. For[3] $r \in \mathcal{B} - k$, let $\mathbf{L}^{kr} \equiv \mathbf{p}^{kN_{kr}}$, and let $\mathbf{L}^{kk} \equiv 0$.

Def. 34. Define $\mathbf{D}^{kk} \equiv -\sum_{j \in \mathcal{B}} \mathbf{L}^{kj} \mathcal{M}_j / \mathcal{M}$ for $k \in \mathcal{B}$.

Def. 35. Let b_k be a point fixed in $\mathscr{b}_k$ such that $\mathbf{D}^{kk}$ is the position vector of o_k with respect to b_k. (This point b_k is called the *barycenter* of the kth substructure in the nominal state.)

Def. 36. Define $\{\mathbf{b}^k\}^T D^{kj} \equiv \mathbf{D}^{kj} \equiv \mathbf{D}^{kk} + \mathbf{L}^{kj}$ for $k, j \in \mathcal{B}$.

Def. 37. Define the dyadic

$$\mathbf{K}^k \equiv \sum_{r \in \mathcal{B}} \mathcal{M}_r (\mathbf{D}^{kr} \cdot \mathbf{D}^{kr} \mathbf{U} - \mathbf{D}^{kr} \mathbf{D}^{kr})$$

---

[3] For notational brevity, the set $\mathcal{B} - \{k\}$ is designated $\mathcal{B} - k$.

where $\mathbf{U}$ is the unit dyadic, and define the corresponding matrix $K^k \equiv \{\mathbf{b}^k\} \cdot \mathbf{K}^k \cdot \{\mathbf{b}^k\}^T$.

Def. 38. Define

$$\mathbf{\Phi}^{kk} \equiv \mathbf{K}^k + \mathbf{J}^k \quad \text{and} \quad \Phi^{kk} \equiv \{\mathbf{b}^k\} \cdot \mathbf{\Phi}^{kk} \cdot \{\mathbf{b}^k\}^T$$

Def. 39. Define

$$\mathbf{\Phi}^{kj} \equiv -\mathfrak{M}(\mathbf{D}^{jk} \cdot \mathbf{D}^{kj}\mathbf{U} - \mathbf{D}^{jk}\mathbf{D}^{kj})$$

with

$$\{\mathbf{b}^j\} \cdot \mathbf{\Phi}^{kj} \cdot \{\mathbf{b}^k\}^T = -\mathfrak{M}(C^{jk}D^{jk^T}C^{jk}D^{kj} - D^{jk}D^{kj^T})$$

Def. 40. Let $\boldsymbol{\omega}^k = \{\mathbf{b}^k\}^T\omega^k$ be the inertial angular velocity of $\mathscr{b}_k$.

Def. 41. Let $\mathbf{h}^k$ be the contribution of rotors in $\mathscr{b}_k$ to the angular momentum of the kth substructure relative to $\mathscr{b}_k$ with respect to o_k, and let $h^k \equiv \mathbf{h}^k \cdot \{\mathbf{b}^k\}$.

Def. 42. Let $\mathfrak{B}_r$ be the rth *neighbor set* for $r \in \mathfrak{B}$, such that $k \in \mathfrak{B}_r$ if $\mathscr{b}_k$ is attached to $\mathscr{b}_r$.

Def. 43. Let $\mathfrak{B}_{jk}$ be the *branch set* of integers r such that $r \in \mathfrak{B}_{jk}$ if $k = N_{jr}$. Thus, $\mathfrak{B}_{jk}$ consists of the indices of those bodies attached to $\mathscr{b}_j$ on a branch which begins with $\mathscr{b}_k$.

Def. 44. Let the tilde symbol ($\tilde{\ }$) signify, in application to a 3 by 1 matrix V with elements V_θ ($\theta = 1, 2, 3$), transformation to a skew-symmetric 3 by 3 matrix $\tilde{V}$ given by

$$\tilde{V} \equiv \begin{bmatrix} 0 & -V_3 & V_2 \\ V_3 & 0 & -V_1 \\ -V_2 & V_1 & 0 \end{bmatrix}$$

B. The Equations

The objective of this section is to begin with the general vector-dyadic equations derived in Ref. 4 and to proceed by sacrificing some of their generality in favor of a *particular* appendage model. Explicit results, in the form of both vector and matrix equations suitable for computer programming, will thereby be obtained.

In what follows, attention is confined to a special case of the finite element appendage model of Ref. 1, for which, as in Ref. 2, all mass of appendage k is concentrated in the n_k discrete nodal bodies of the appendage (with no distributed mass for the internodal elastic elements). All deformations from a nominal appendage state are assumed arbitrarily small, so that terms above the first degree in these deformations (and corresponding rates) can be neglected. Further, any rigid body $\mathscr{b}_k$ will be assumed to carry rotors, and they will consist of an orthogonal triad whose axes parallel $\mathbf{b}_1^k$, $\mathbf{b}_2^k$, and $\mathbf{b}_3^k$.

The starting point for this development is the set of vector-dyadic equations of vehicle translation and substructure rotation as derived in Ref. 4 (Eqs. 9, 31–35):

$$\mathbf{F} = \mathfrak{M}\ddot{\mathbf{X}} \tag{1}$$

$$\sum_{k \in \mathfrak{B}} \mathbf{W}^k = 0 \tag{2}$$

$$\tau_s + \mathbf{g}^s \cdot \sum_{k \in \mathcal{P}} \epsilon_{sk} \mathbf{W}^k = 0 \qquad (s \in \mathcal{P}) \tag{3}$$

where

$$\begin{aligned}
\mathbf{W}^k \equiv\; & \mathbf{T}^k + \sum_{r \in \mathfrak{B}} \mathbf{D}^{kr} \times \mathbf{F}^r + c^k \times \left(\frac{\mathfrak{M}_k}{\mathfrak{M}} \mathbf{F} - \mathbf{F}^k \right) \\
& + \sum_{r \in \mathfrak{B}} \mathfrak{M}_r \mathbf{D}^{kr} \times [\ddot{\mathbf{c}}^r + 2\boldsymbol{\omega}^r \times \dot{\mathbf{c}}^r + \dot{\boldsymbol{\omega}}^r \times \mathbf{c}^r + \boldsymbol{\omega}^r \times (\boldsymbol{\omega}^r \times \mathbf{c}^r)] \\
& + \mathfrak{M}_k \mathbf{c}^k \times \sum_{r \in \mathfrak{B}} [\dot{\boldsymbol{\omega}}^r \times \mathbf{D}^{rk} + \boldsymbol{\omega}^r \times (\boldsymbol{\omega}^r \times \mathbf{D}^{rk})] \\
& - \boldsymbol{\Phi}^{kk} \cdot \dot{\boldsymbol{\omega}}^k - \sum_{r \in \mathfrak{B} - k} \boldsymbol{\Phi}^{kr} \cdot \dot{\boldsymbol{\omega}}^r + \mathfrak{M} \sum_{r \in \mathfrak{B} - k} \mathbf{D}^{kr} \times [\boldsymbol{\omega}^r \times (\boldsymbol{\omega}^r \times \mathbf{D}^{rk})] \\
& - \boldsymbol{\omega}^k \times \boldsymbol{\Phi}^{kk} \cdot \boldsymbol{\omega}^k - \dot{\mathbf{h}}^k - \boldsymbol{\omega}^k \times \mathbf{h}^k - \dot{\boldsymbol{\Phi}}^{kk} \cdot \boldsymbol{\omega}^k \\
& - \int_{a_k} \mathbf{p} \times \ddot{\mathbf{p}}\, dm - \boldsymbol{\omega}^k \times \int_{a_k} (\mathbf{p} \times \dot{\mathbf{p}})\, dm
\end{aligned} \tag{4}$$

and

$$\boldsymbol{\omega}^k = \boldsymbol{\omega}^0 + \sum_{r \in \mathcal{P}} \epsilon_{rk} \dot{\gamma}_r \mathbf{g}^r \tag{5}$$

$$\dot{\boldsymbol{\omega}}^k = \dot{\boldsymbol{\omega}}^0 + \sum_{r \in \mathcal{P}} \epsilon_{rk} [\ddot{\gamma}_r \mathbf{g}^r + \boldsymbol{\omega}^r \times \mathbf{g}^r \dot{\gamma}_r] \tag{6}$$

The adoption of a nodal body appendage model leads (as in Ref. 2, Eq. 58) to the following useful relation:

$$\mathbf{c}^k = -\sum_{s=1}^{n_k} \frac{m_s}{\mathfrak{M}_k} \mathbf{u}^s \tag{7}$$

where appendage a_k has been idealized as n_k nodal bodies interconnected by massless elastic structure, with m_s the mass of nodal body s, and $\mathbf{u}^s$ the displacement of the body s relative to b_k from the position occupied in the nominal state.

It will also be necessary to develop an expression for $\dot{\boldsymbol{\Phi}}^{kk}$ in terms of appendage variables. From Def. 38, we know that

$$\Phi^{kk} = \mathbf{K}^k + \mathbf{J}^k \tag{8}$$

where $\mathbf{K}^k$, the "augmented" inertia dyadic, is a constant. $\mathbf{J}^k$, the inertia dyadic of the kth substructure for o_k, is time-variable due to appendage deformations and may be obtained from

$$\mathbf{J}^k = \int (\mathbf{p} \cdot \mathbf{p}\mathbf{U} - \mathbf{p}\mathbf{p}) dm \tag{9}$$

where $\mathbf{U}$ is the unit dyadic.

For the small-deformation appendage model adopted here, $\mathbf{J}^k$ may be evaluated (see Ref. 2, Eq. 126) as

$$\mathbf{J}^k = \bar{\mathbf{J}}^k + \{\mathbf{b}^k\}^T \left[\sum_{s=1}^{n_k} \left\{ m_s [2(R^k + r^s)^T u^s U - (R^k + r^s) u^{s^T} - u^s (R^k + r^s)^T] + \tilde{\beta}^s I^s - I^s \tilde{\beta}^s \right\} \right] \{\mathbf{b}^k\} \tag{10}$$

where $\bar{\mathbf{J}}^k$ is the nominal (constant) value of $\mathbf{J}^k$, and I^s is the constant inertia matrix of the sth nodal body for its own mass center and in its own body-fixed vector basis $\{\mathbf{n}^s\}^k$, where in the nominal state, $\{\mathbf{n}^s\}^k = \{\mathbf{b}^k\}$.

Combining (8) and (10), we have

$$\dot{\Phi}^{kk} = \{\mathbf{b}^k\}^T \left[\sum_{s=1}^{n_k} \left\{ m_s [2(R^k + r^s)^T \dot{u}^s U - (R^k + r^s) \dot{u}^{s^T} - \dot{u}^s (R^k + r^s)^T] + \tilde{\dot{\beta}}^s I^s - I^s \tilde{\dot{\beta}}^s \right\} \right] \{\mathbf{b}^k\} \tag{11}$$

Finally, Eq. (4) requires more explicit expressions for the integrals over the appendage a_k. The appropriate expressions in this case may be found in Eq. (114) of Ref. 2, which simplifies to

$$-\frac{^i d}{dt} \int_{a_k} \mathbf{p} \times \mathring{\mathbf{p}}\, dm = -\int_{a_k} \mathbf{p} \times \mathring{\mathring{\mathbf{p}}}\, dm - \boldsymbol{\omega}^k \times \int_{a_k} \mathbf{p} \times \mathring{\mathbf{p}}\, dm$$

or

$$-\frac{^i d}{dt} \int_{a_k} \mathbf{p} \times \mathring{\mathbf{p}}\, dm = -\sum_{s=1}^{n_k} (\mathbf{R}^k + \mathbf{r}^s) \times m_s \mathring{\mathring{\mathbf{u}}}^s - \boldsymbol{\omega}^k \times \sum_{s=1}^{n_k} (\mathbf{R}^k + \mathbf{r}^s) \times m_s \mathring{\mathbf{u}}^s - \sum_{s=1}^{n_k} (\mathbf{I}^s \cdot \mathring{\mathring{\boldsymbol{\beta}}}^s + \boldsymbol{\omega}^k \times \mathbf{I}^s \cdot \mathring{\boldsymbol{\beta}}^s) \tag{12}$$

Note that in Eqs. (7), (10), (11), and (12), the superscript k has been dropped from nodal body variables in the kth appendage (such as u^s, which replaces $(u^s)^k$).

Turning now to the appendage equations, we will make use of the nodal body finite element model case described by Eq. (95) of Ref. 2 (correcting the last algebraic sign within the braces on the right side of Eq. 95 by changing − to +, and subtracting all nominally nonzero terms from the right side so as to make q a measure of the deviation from a nominal state in which the appendage might be deformed). In matrix form, the equation for the kth appendage becomes

$$M^k\left(U - \Sigma_{U0}\Sigma_{U0}^T \frac{M^k}{\mathfrak{M}_k}\right)\ddot{q}^k + \left\{2M^k\left[(\Sigma_{U0}\omega^k)^\sim - \Sigma_{U0}\tilde{\omega}^k\Sigma_{U0}^T \frac{M^k}{\mathfrak{M}_k}\right]\right.$$

$$\left. + M^k(\Sigma_{0U}\omega^k)^\sim + (\Sigma_{0U}\omega^k)^\sim M^k - (M^k\Sigma_{0U}\omega^k)^\sim \right\}\dot{q}^k$$

$$+ \left\{ M^k(\Sigma_{0U}\dot{\omega}^k)^\sim - (M^k\Sigma_{0U}\dot{\omega}^k)^\sim - (\Sigma_{0U}\omega^k)^\sim(M^k\Sigma_{0U}\omega^k)^\sim \right.$$

$$+ (\Sigma_{0U}\omega^k)^\sim M^k(\Sigma_{0U}\omega^k)^\sim + M^k\left[(\Sigma_{U0}\dot{\omega}^k)^\sim\right.$$

$$\left.\left. - \Sigma_{U0}(\tilde{\dot{\omega}}^k + \tilde{\omega}^k\tilde{\omega}^k)\Sigma_{U0}^T \frac{M^k}{\mathfrak{M}_k} + (\Sigma_{U0}\omega^k)^\sim(\Sigma_{U0}\omega^k)^\sim\right] + K^k\right\}q^k$$

$$= -M^k\left\{\Sigma_{0U}\dot{\omega}^k + \Sigma_{U0}[\Theta\ddot{X} - \tilde{R}^k\dot{\omega}^k + \tilde{\omega}^k\tilde{\omega}^k R^k - \tilde{\Omega}^k\tilde{\Omega}^k R^k]\right.$$

$$+ (\Sigma_{U0}\omega^k)^\sim(\Sigma_{U0}\omega^k)^\sim r_k - (\Sigma_{U0}\Omega^k)^\sim(\Sigma_{U0}\Omega^k)^\sim r_k$$

$$\left. - \tilde{r}_k\Sigma_{U0}\dot{\omega}^k\right\} - (\Sigma_{0U}\omega^k)^\sim M^k(\Sigma_{0U}\omega^k) + (\Sigma_{0U}\Omega^k)^\sim M^k(\Sigma_{0U}\Omega^k) + \lambda^k \qquad (13)$$

where the assumption has been made that the appendage structure contains no damping. The symbol λ^k is a column matrix containing any forces or torques applied to the n_k sub-bodies of the appendage other than the structural interaction forces induced by deformations. For example, gravity forces or attitude control jet thrust would contribute to λ^k. Also,

$$q^k \equiv [u_1^1 u_2^1 u_3^1 \beta_1^1 \beta_2^1 \beta_3^1 u_1^2 \cdots \beta_3^{n_k}]^T$$

a $6n_k$ by 1 matrix which fully characterizes the appendage deformations relative to some nominal state of deformation induced by the nominal constant value Ω^k of ω^k.

$$
M^k \equiv \begin{bmatrix} m^1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & I^1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & m^2 & 0 & \cdots & 0 \\ 0 & 0 & 0 & I^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots\vdots\vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & I^{n_k} \end{bmatrix}
$$

a constant, symmetric $6n_k$ by $6n_k$ matrix defined in terms of the 3 by 3 partitioned matrices m^s, I^s.

$$
m^s \equiv \begin{bmatrix} m_s & 0 & 0 \\ 0 & m_s & 0 \\ 0 & 0 & m_s \end{bmatrix}, \quad I^s = \begin{bmatrix} I^s_{11} & I^s_{12} & I^s_{13} \\ I^s_{21} & I^s_{22} & I^s_{23} \\ I^s_{31} & I^s_{32} & I^s_{33} \end{bmatrix} \quad (s = 1, \ldots n_k)
$$

$$
\Sigma_{U0} = [U \quad 0 \quad U \quad 0 \quad \cdots \quad U \quad 0]^T
$$

$$
\Sigma_{0U} = [0 \quad U \quad 0 \quad U \quad \cdots \quad 0 \quad U]^T
$$

$6n_k$ by 3 Boolean operator matrices, where U and 0 are the 3 by 3 unit and null matrices, respectively.

$$
r_k \equiv [r^{1^T} \quad 0 \quad r^{2^T} \quad 0 \quad \cdots \quad r^{n_k^T} \quad 0]^T
$$

$$
\tilde{r}_k \equiv \begin{bmatrix} \tilde{r}^1 & 0 & 0 & 0 & \cdots & 0 & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \tilde{r}^2 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots\vdots\vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \tilde{r}^{n_k} & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}
$$

a constant $6n_k$ by $6n_k$ matrix.

$K^k \equiv$ the stiffness matrix that determines the structural interaction forces and torques induced by deformation of the kth appendage from its nominal state (a constant, symmetric $6n_k$ by $6n_k$ matrix).

It should now be recognized that the term $\Theta\ddot{X}$ in Eq. (13) must be replaced by the inertial acceleration of the mass center of the corresponding substructure in the local vector basis, which is assumed for each k to be zero in the "nominal" state. For substructure s_k, this term is given by (see Eq. 54, Ref. 4)

$$\Theta\ddot{X} = C^{k0}C\ddot{X} - (\ddot{c}^k + \dot{\tilde{\omega}}^k c^k + 2\tilde{\omega}^k \dot{c}^k + \tilde{\omega}^k \tilde{\omega}^k c^k)$$

$$+ \sum_{r \in \mathfrak{B}} C^{kr}\left[(\dot{\tilde{\omega}}^r + \tilde{\omega}^r\tilde{\omega}^r)\left(D^{rk} + \frac{\mathfrak{M}_r}{\mathfrak{M}} c^r\right) + \frac{\mathfrak{M}_r}{\mathfrak{M}} (\ddot{c}^r + 2\tilde{\omega}^r \dot{c}^r)\right] \tag{14}$$

and

$$C\ddot{X} = \frac{F}{\mathfrak{M}} \tag{15}$$

treated as zero in the nominal state.

Equations (1)–(15) provide a rather complete system description (although the contribution of rigid rotors, i.e., $\mathbf{h}^k$, will be developed in more detail later). Since the number of nodes n_k in a single finite-element model of an elastic appendage is typically rather large, it is to be understood that the nodal body vibration equations, (Eqs. 13–15), will provide the basis for a transformation to distributed or modal coordinates for appendage deformations and that most of these will be deleted from consideration by truncating the matrix of deformation variables. Thus, the variables labeled u^s and β^s above will be replaced by appropriate combinations of new modal deformation variables.

The equations actually to be programmed for digital computer solution will therefore be the transformed and truncated versions of Eqs. (1)–(15). These will be described in the following sections as the system motions are confined to two particular cases of interest: (1) the case in which all appendage base-body angular rates ω^k experience only slight deviations from some constant nonzero value (i.e., $\omega^k \approx \Omega^k$, $\dot{\omega}^k \approx 0$), or (2) the case in which $\Omega^k \approx 0$ (i.e., $\omega^k \approx 0$, $\dot{\omega}^k \approx 0$) for all appendage base bodies.

In the first case, i.e., where $\omega^k \approx \Omega^k$ and $\dot{\omega}^k \approx 0$, the approach taken in developing the system equations of motion, including linearization, coordinate transformation, and truncation, may be described as follows:

(1) For the purposes of constructing a coordinate transformation for the appendages, assume that ω^k experiences only small deviations from a constant Ω^k, and write the homogeneous form of the appendage equations.

(2) Construct a coordinate transformation from these linear, constant-coefficient equations, and select the truncation level.

(3) Return to the unrestricted ω^k assumption, and substitute the transformations from (2) into *all* equations of motion.

(4) In the homogeneous part of the appendage vibration equations *only*, ignore products of deformation variables and deviations of ω^k from Ω^k. *This step is not formally correct*, since mathematically we cannot justify treating the deviation of ω^k from Ω^k as small only when it is multiplied by a deformation variable. On the basis of engineering judgment, however, the authors feel that it is probably justifiable and would be a less significant source of error than either modeling or truncation. The resulting equations contain all terms formally required for the analysis of a system with appendage base bodies experiencing small deviations from their nominal motions, but in applying these equations to systems with large deviations of base bodies from their

nominal motion one is suppressing products of these deviations with deformation variables. In fact, a very large change in base-body spin rate would change the effective structural stiffness of the appendage, and invalidate the modal analysis on which the appendage modal coordinate selection is based. In this respect, the equations would be tainted by truncation even if the suppressed terms were retained, and, since these terms would substantially complicate the analysis by coupling all variables into each vibration equation, they have been rejected here.

N76 12093

III. Systems With Rotating Appendages

A. Equations

Inspection of the appendage equations (Eqs. 13–15) reveals that the coefficients of q^k and $\dot{q}^k$ depend upon ω^k, which characterizes the rotational motion of the appendage base. In general, ω^k is an unknown function of time, to be determined only after the appendage equations are augmented by other equations of dynamics and control for the total vehicle and solved. Only if ω^k can be assumed to experience, in a given time interval, small excursions about a constant nominal value Ω^k is there any possibility of transforming Eq. (13) to a new set of *uncoupled* appendage coordinates. Any methods involving modal coordinates (see Ref. 1, Sect. I) depend formally upon this assumption.

Assuming then that $\omega^k \approx \Omega^k$ and $\dot{\omega}^k \approx 0$, Eqs. (13)–(15) can be combined to provide the following appendage equation:

$$
\begin{aligned}
& M^k\left(U - \Sigma_{U0}\Sigma_{U0}^T \frac{M^k}{\mathfrak{M}}\right)\ddot{q}^k + \Bigg\{2M^k\left[(\Sigma_{U0}\Omega^k)^{\sim} - \Sigma_{U0}\tilde{\Omega}^k\Sigma_{U0}^T \frac{M^k}{\mathfrak{M}}\right] \\
&\quad + M^k(\Sigma_{0U}\Omega^k)^{\sim} + (\Sigma_{0U}\Omega^k)^{\sim} M^k - (M^k\Sigma_{0U}\Omega^k)^{\sim}\Bigg\}\dot{q}^k \\
&\quad + \Bigg\{-(\Sigma_{0U}\Omega^k)^{\sim}(M^k\Sigma_{0U}\Omega^k)^{\sim} + (\Sigma_{0U}\Omega^k)^{\sim} M^k(\Sigma_{0U}\Omega^k)^{\sim} \\
&\quad + M^k\left[-\Sigma_{U0}(\tilde{\Omega}^k\tilde{\Omega}^k)\Sigma_{U0}^T \frac{M^k}{\mathfrak{M}} + (\Sigma_{U0}\Omega^k)^{\sim}(\Sigma_{U0}\Omega^k)^{\sim}\right] + K^k\Bigg\}q^k \\
&= \left(-M^k\Sigma_{0U} + M^k\Sigma_{U0}\tilde{R}^k + M^k\tilde{r}_k\Sigma_{U0}\right)\dot{\omega}^k - M^k\Sigma_{U0}\sum_{r\in\mathfrak{B}} C^{kr}(\dot{\tilde{\omega}}^r + \tilde{\omega}^r\tilde{\omega}^r)D^{rk} \\
&\quad - M^k\left[\Sigma_{U0}C^{k0}\frac{F}{\mathfrak{M}} + \Sigma_{U0}\tilde{\omega}^k\tilde{\omega}^k R^k + (\Sigma_{U0}\omega^k)^{\sim}(\Sigma_{U0}\omega^k)^{\sim} r_k\right] \\
&\quad - (\Sigma_{0U}\omega^k)^{\sim} M^k(\Sigma_{0U}\omega^k) + \lambda^k + M^k\left[\Sigma_{U0}\tilde{\Omega}^k\tilde{\Omega}^k R^k + (\Sigma_{U0}\Omega^k)^{\sim}(\Sigma_{U0}\Omega^k)^{\sim} r_k\right] \\
&\quad + (\Sigma_{0U}\Omega^k)^{\sim} M^k(\Sigma_{0U}\Omega^k) \\
&\quad - M^k\Sigma_{U0}\sum_{r\in\mathfrak{B}-k} C^{kr}\left[-\Sigma_{U0}^T\frac{M^r}{\mathfrak{M}}\ddot{q}^r + 2\tilde{\omega}^r\frac{\mathfrak{M}_r}{\mathfrak{M}}\dot{c}^r + \tilde{\omega}^r\tilde{\omega}^r\frac{\mathfrak{M}_r}{\mathfrak{M}}c^r\right] \qquad (16)
\end{aligned}
$$

Equation (16) consists of $6n_k$ second-order scalar equations and can be written as a matrix equation with the following structure:

$$M_k'\ddot{q}^k + D_k'\dot{q}^k + G_k'\dot{q}^k + K_k'q^k + A_k'q^k = L_k' \tag{17}$$

where

$$M_k' = M^k\left(U - \Sigma_{U0}\Sigma_{U0}^T \frac{M^k}{\mathfrak{M}}\right)$$

$$D_k' = 0$$

$$G_k' = 2M^k\left[(\Sigma_{U0}\Omega^k)^\sim - \Sigma_{U0}\tilde{\Omega}^k\Sigma_{U0}^T \frac{M^k}{\mathfrak{M}}\right] + M^k(\Sigma_{0U}\Omega^k)^\sim$$

$$+ (\Sigma_{0U}\Omega^k)^\sim M^k - (M^k\Sigma_{0U}\Omega^k)^\sim$$

$$A_k' = -(\Sigma_{0U}\Omega^k)^\sim(M\Sigma_{0U}\Omega^k)^\sim$$

$$K_k' = (\Sigma_{0U}\Omega^k)^\sim M^k(\Sigma_{0U}\Omega^k)^\sim + K^k$$

$$+ M^k\left[-\Sigma_{U0}(\tilde{\Omega}^k\tilde{\Omega}^k)\Sigma_{U0}^T \frac{M^k}{\mathfrak{M}} + (\Sigma_{U0}\Omega^k)^\sim(\Sigma_{U0}\Omega^k)^\sim\right]$$

and

$$L_k' = -M^k\left[\Sigma_{0U} - \Sigma_{U0}(\tilde{R}^k + \tilde{D}^{kk}) - \tilde{r}_k\Sigma_{U0}\right]\dot{\omega}^k - M^k\Sigma_{U0}C^{k0}\frac{F}{\mathfrak{M}} + \lambda^k$$

$$-M^k\Sigma_{U0}\sum_{r\in\mathfrak{B}-k} C^{kr}(\tilde{\dot{\omega}}^r + \tilde{\omega}^r\tilde{\omega}^r)D^{rk} + N_k^c - N_{k_\infty}^c$$

$$-M^k\Sigma_{U0}\sum_{r\in\mathfrak{B}-k} C^{kr}\left[-\Sigma_{U0}^T\frac{M^r}{\mathfrak{M}}\ddot{q}^r + 2\tilde{\omega}^r\frac{\mathfrak{M}_r}{\mathfrak{M}}\dot{c}^r + \tilde{\omega}^r\tilde{\omega}^r\frac{\mathfrak{M}_r}{\mathfrak{M}}c^r\right]$$

with

$$N_k^c = -M^k[\Sigma_{U0}\tilde{\omega}^k\tilde{\omega}^k(R^k + D^{kk})$$

$$+ (\Sigma_{U0}\omega^k)^\sim(\Sigma_{U0}\omega^k)^\sim r_k] - (\Sigma_{0U}\omega^k)^\sim M^k(\Sigma_{0U}\omega^k)$$

and

$$N_{k_\infty}^c = -M^k\left[\Sigma_{U0}\tilde{\Omega}^k\tilde{\Omega}^k(R^k + D^{kk}) + (\Sigma_{U0}\Omega^k)^\sim(\Sigma_{U0}\Omega^k)^\sim r_k\right] - (\Sigma_{0U}\Omega^k)^\sim M^k(\Sigma_{0U}\Omega^k)$$

Matrices M_k', D_k', and K_k' are constant symmetric matrices, while G_k' is a constant skew-symmetric matrix, and A_k' has both symmetric and skew-symmetric parts. N_k^c

contains the nonlinear terms in ω^k due to centripetal accelerations of the appendage due to ω^k, and $N^c_{k_{ss}}$ represents the nominal steady-state value of N^c_k.

Notice that the form of Eq. (17) is identical to that of Eq. (140) in Ref. 2 (or Eq. 64, Ref. 1), with the exception of the additional right-hand-side terms

$$-M^k\Sigma_{U0}\sum_{r\in\mathfrak{B}-k} C^{kr}\left[-\Sigma^T_{U0}\frac{M^r}{\mathfrak{M}}\ddot{q}^r + 2\tilde{\omega}^r\frac{\mathfrak{M}_r}{\mathfrak{M}}\dot{c}^r + \tilde{\omega}^r\tilde{\omega}^r\frac{\mathfrak{M}_r}{\mathfrak{M}}c^r + (\dot{\tilde{\omega}}^r + \tilde{\omega}^r\tilde{\omega}^r)D^{rk}\right]$$

which describe the coupling of appendage a_k to other rigid bodies and appendages of the system. Also, in comparing Eq. (17) to Eq. (140) of Ref. 2, note that **R** has been replaced by $(\mathbf{R}^k + \mathbf{D}^{kk})$, a vector from the mass center (barycenter) of the undeformed *augmented* substructure to the point Q_k (see Fig. 1 and Def. 35).

At this point in the development of the appendage equations, it is appropriate to elaborate upon what is meant by "nominal appendage state," and what relationship this idea has to Eq. (17). We have already indicated that the approach to be taken is that of Ref. 1 (see pp. 1–3), namely that appendages are ideally considered as linearly elastic and that **u** and $\boldsymbol{\beta}$ are "small," oscillatory appendage deformations, i.e., variational deformations. It is quite possible that these small oscillatory deformations will be superimposed on relatively large steady-state deformations, due to spin, for example.

In order to derive a suitable appendage equation, applicable for a "variational deformation" **q**, the substitution of an expansion for the total deformation **q**′ such as

$$\mathbf{q}' = \mathbf{q} + \mathbf{q}_{ss}$$

has been made in Eq. (17), where $\mathbf{q}_{ss}$ (= constant) is understood to be the steady-state appendage deformation due to spin. The steady-state deformation is given by

$$(K'_k + A'_k)\mathbf{q}_{ss} = N^c_{k_{ss}}$$

where

$$N^c_{k_{ss}} = -M^k\left[\Sigma_{U0}\tilde{\Omega}^k\tilde{\Omega}^k(R^k + D^{kk}) + (\Sigma_{U0}\Omega^k)^{\sim}(\Sigma_{U0}\Omega^k)^{\sim}r_k\right]$$

$$-(\Sigma_{0U}\Omega^k)^{\sim}M^k(\Sigma_{0U}\Omega^k)$$

In effect then, in Eq. (17), we have linearized about the steady-state deformation induced by centrifugal forces due to spin of the kth substructure, with the mass center of this substructure inertially fixed. It should also be remembered that the original definitions of o_k, c_k, and the vectors $(\mathbf{r})^k$, $\mathbf{R}^k$, etc., remain intact but that the term "nominal state" is more clearly specified as the "steady state" of deformation due to the nominal (constant) spin of the kth substructure, with the mass center of that substructure inertially fixed. Also, the value of K^k should include whatever increment to the elastic stiffness of the appendage is attributable to structural preload due to this spin; that is, K^k includes the so-called "geometric stiffness matrix" of the structure.

The matrix D', which in the general case would accommodate any viscous damping that may be introduced to represent energy dissipation due to structural vibrations, is zero here since such terms have been omitted. But they can still be inserted if one accepts the practice common among structural dynamicists of incorporating the equivalent of a term $D'_k \dot{q}^k$ into equations of vibration only *after* derivation of equations of motion and transformation of coordinates.

The nature of terms contributing to G'_k, K'_k, and A'_k is discussed in some detail in Ref. 1. In particular, the matrix A'_k is shown in Ref. 1 to disappear for the case of small base excursions about a nonzero constant spin only if the nodal bodies are particles or spheres, *or* if in the steady state of deformation, all nodal bodies have principal axes of inertia aligned with the nominal value of the angular velocity $\boldsymbol{\omega}^k$ (i.e., $\boldsymbol{\omega}^k \approx \{\mathbf{b}^k\}\Omega^k$). The *latter* restriction will henceforth be adopted in this report since it greatly reduces the computational task in transforming the homogeneous form of Eq. (17) to a set of completely uncoupled differential equations.

In order to transform Eq. (17) to a set of uncoupled equations, it is first necessary to rewrite it in first-order form, such as

$$\mathcal{U}_k \dot{Q}^k + \mathcal{V}_k Q^k = \mathcal{L}_k \tag{18}$$

where

$$Q^k \equiv \begin{bmatrix} q^k \\ \hline \dot{q}^k \end{bmatrix} \qquad \mathcal{L}_k \equiv \begin{bmatrix} 0 \\ \hline L'_k \end{bmatrix}$$

$$\mathcal{U}_k \equiv \left[\begin{array}{c|c} K'_k & 0 \\ \hline 0 & M'_k \end{array}\right] \qquad \mathcal{V}_k \equiv \left[\begin{array}{c|c} 0 & -K'_k \\ \hline K'_k & G'_k \end{array}\right]$$

Now let Φ be a $(12n_k \times 12n_k)$ matrix of (complex) eigenvectors of the differential operator in Eq. (18), and let Φ' be a $(12n_k \times 12n_k)$ matrix of (complex) eigenvectors of the homogeneous adjoint equation

$$\mathcal{U}_k^T \dot{Q}'^k + \mathcal{V}_k^T Q'^k = 0 \tag{19}$$

so that Φ_k and Φ'_k are related by

$$\Phi_k^{-1} = l \Phi_k'^T$$

with l a $(12n_k \times 12n_k)$ diagonal matrix which depends upon the normalization of Φ_k and Φ'_k. Substitution into Eq. (18) of the transformation

$$Q^k = \Phi_k Y^k$$

and premultiplication by $\Phi_k'^T$ furnishes

$$(\Phi_k'^T \mathcal{U}_k \Phi_k) \dot{Y}^k + (\Phi_k'^T \mathcal{V}_k \Phi_k) Y^k = \Phi_k'^T \mathcal{L}_k$$

The two coefficient matrices enclosed in parentheses are diagonal. If Λ_k is the $(12n_k \times 12n_k)$ matrix of the (complex) eigenvalues of the differential operator in Eq. (18), then upon premultiplication by $(\Phi_k'^T \mathcal{U}_k \Phi_k)^{-1}$, one obtains

$$\dot{Y}^k = \Lambda_k Y^k + \left(\Phi_k'^T \mathcal{U}_k \Phi_k\right)^{-1} \Phi_k'^T \mathcal{L}_k \tag{20}$$

Note that the matrix inversion in Eq. (20) consists simply of calculating the reciprocals of the diagonal elements of $\Phi_k'^T \mathcal{U}_k \Phi_k$.

In practice, one may expect that physical interpretation of the new (complex) state variables $Y_1^k, \ldots, Y_{12n_k}^k$ will permit truncation to a reduced set of variables

$$\bar{Y}^k \equiv \left[Y_1^k \cdots Y_{N_k}^k Y_1^{k*} \cdots Y_{N_k}^{k*} \right]^T \tag{21}$$

where N_k is the number of modes to be preserved in the simulation. The transformation matrix Φ_k is accordingly truncated to the $(12n_k \times 2N_k)$ matrix $\bar{\Phi}_k$, where

$$\bar{\Phi}_k \equiv [\Phi_k' \cdots \Phi_k^{N_k} \Phi_k'^* \cdots \Phi_k^{N_k *}]$$

The equation of motion of the appendage now becomes

$$\dot{\bar{Y}}^k = \begin{bmatrix} \lambda_1 & & & & & 0 \\ & \ddots & & & & \\ & & \lambda_{N_k} & & & \\ & & & \lambda_1^* & & \\ & 0 & & & \ddots & \\ & & & & & \lambda_{N_k}^* \end{bmatrix} \bar{Y}^k + \left(\bar{\Phi}_k'^T \mathcal{U}_k \bar{\Phi}_k\right)^{-1} \bar{\Phi}_k'^T \mathcal{L}_k \tag{22}$$

Since, in the particular case studied here, the matrices $\mathcal{U}_k$ and $\mathcal{V}_k$ in Eq. (18) are, respectively, symmetric and skew-symmetric, so that Eq. (19) becomes

$$\mathcal{U}_k \dot{Q}'^k - \mathcal{V}_k Q'^k = 0 \tag{23}$$

the adjoint eigenvector matrix is available as the complex conjugate

$$\Phi_k' = \Phi_k^* \tag{24}$$

The final equations, after truncation of Eq. (24) and substitution into (22), are therefore obtained without the necessity of actually computing the eigenvectors constituting Φ'^k. Thus, Eq. (22) becomes

$$\dot{\bar{Y}}^k = \bar{\Lambda}_k \bar{Y}^k + \left(\bar{\Phi}_k^{*T} \mathcal{U}_k \bar{\Phi}_k\right)^{-1} \bar{\Phi}_k^{*T} \mathcal{L}_k \tag{25}$$

Since the appendage modeling process thus far has assumed that the structure contains no damping, the diagonal matrix, Λ_k, will contain only eigenvalues that are purely imaginary, e.g., $\lambda_m = \pm i\sigma_m$. Conventional practice in structural dynamics, if some energy dissipation in the model is desired, is to rather arbitrarily add what amounts to a viscous damping term $D_k'\dot{q}^k$ to the appendage equation *after* completing the modal analysis, assuming that the structure of D_k' is such that eigenvectors $\Phi_k^1, \ldots, \Phi_k^{12n_k}$ are undisturbed by this addition. Specifically, one substitutes $\lambda_m = -\xi_m\sigma_m \pm i\sigma_m$ into Eq. (22) or (25), where ξ_m is the "percent of critical damping" and is chosen based on experience (including tests) with similar structures. (See Appendix A for a discussion of some ramifications of adding damping *after* transforming the appendage equations to modal coordinates.)

An apparent disadvantage of Eq. (25) is the fact that the quantities $\overline{Y}^k$, $\overline{\Lambda}_k$, and $\overline{\Phi}_k$ are complex. However, Eq. (25) can be written in terms of its real and imaginary parts and the resulting equations greatly simplified by the use of certain orthogonality relationships. The detailed development of the equations is shown in Ref. 3, and only the results are presented here.

Realizing that Φ_k^j must have the form

$$\Phi_k^j = \begin{bmatrix} \phi^j \\ -- \\ \phi^j\lambda_j \end{bmatrix}, \quad (\Phi_k^j = j\text{th column of } \Phi_k, j = 1, \ldots, 12n_k)$$

where $\phi^j = \psi^j + i\Gamma^j$, $(6n_k \times 1)$, and letting $Y_\alpha^k = \delta_\alpha^k + i\eta_\alpha^k$, $Y_\alpha^{k*} = \delta_\alpha^k - i\eta_\alpha^k$, $(\alpha = 1, \ldots, 6n_k)$, one can see from Ref. 3 that the real $N_k \times 1$ (truncated) matrices, $\bar{\delta}^k$ and $\bar{\eta}^k$, are the solutions to the equations

$$\dot{\bar{\delta}}^k = -\bar{\sigma}^k\bar{\eta}^k - \bar{\sigma}^k\overline{\Gamma}_k^T L_k' - \bar{\xi}^k\bar{\sigma}^k\bar{\delta}^k \tag{26a}$$

and

$$\dot{\bar{\eta}}^k = \bar{\sigma}^k\bar{\delta}^k - \bar{\sigma}^k\bar{\psi}_k^T L_k' - \bar{\xi}^k\bar{\sigma}^k\bar{\eta}^k \tag{26b}$$

As a result, the relationships between the *real* quantities q^k, $\dot{q}^k$, δ^k, and η^k, in matrix terms, are as follows:

$$q^k = 2\left(\bar{\psi}_k\bar{\delta}^k - \overline{\Gamma}_k\bar{\eta}^k\right) \tag{27a}$$

and

$$\dot{q}^k = -2\left(\overline{\Gamma}_k\bar{\sigma}^k\bar{\delta}^k + \bar{\psi}_k\bar{\sigma}^k\bar{\eta}^k\right) \tag{27b}$$

so that

$$\ddot{q}^k = -2\left(\overline{\Gamma}_k\bar{\sigma}^k\dot{\bar{\delta}}^k + \bar{\psi}_k\bar{\sigma}^k\dot{\bar{\eta}}^k\right) \tag{27c}$$

In order to complete the set of model equations, particularly in the form suitable for computer solution, it is necessary to return to the vehicle equations, substituting the relations developed in Eqs. (7), (11), (12), etc., into Eq. (4), to obtain

$$
\begin{aligned}
\mathbf{W}^k = {} & \mathbf{T}^k + \sum_{r\in\mathfrak{B}} \mathbf{D}^{kr} \times \mathbf{F}^r + \mathbf{c}^k \times \left(\frac{\mathfrak{M}_k}{\mathfrak{M}} \mathbf{F} - \mathbf{F}^k \right) \\
& + \sum_{r\in\mathfrak{B}} \mathfrak{M}_r \mathbf{D}^{kr} \times \left[-\sum_{s=1}^{n_r} \frac{m_r}{\mathfrak{M}_r} \mathring{\mathring{\mathbf{u}}}^s + 2\boldsymbol{\omega}^r \times \mathring{\mathbf{c}}^r + \dot{\boldsymbol{\omega}}^r \times \mathbf{c}^r + \boldsymbol{\omega}^r \times (\boldsymbol{\omega}^r \times \mathbf{c}^r) \right] \\
& + \mathfrak{M}_k \mathbf{c}^k \times \sum_{r\in\mathfrak{B}} [\dot{\boldsymbol{\omega}}^r \times \mathbf{D}^{rk} + \boldsymbol{\omega}^r \times (\boldsymbol{\omega}^r \times \mathbf{D}^{rk})] \\
& - \sum_{r\in\mathfrak{B}} \boldsymbol{\Phi}^{kr} \cdot \dot{\boldsymbol{\omega}}^r + \mathfrak{M} \sum_{r\in\mathfrak{B}-k} \mathbf{D}^{kr} \times [\boldsymbol{\omega}^r \times (\boldsymbol{\omega}^r \times \mathbf{D}^{rk})] \\
& - \boldsymbol{\omega}^k \times \boldsymbol{\Phi}^{kk} \cdot \boldsymbol{\omega}^k - \mathring{\mathbf{h}}^k - \boldsymbol{\omega}^k \times \mathbf{h}^k \\
& - \{\mathbf{b}^k\}^T \sum_{s=1}^{n_k} \Big[m_s \big\{ 2(R^k + r^s)^T \dot{u}^s U - (R^k + r^s) \dot{u}^{s^T} - \dot{u}^s (R^k + r^s)^T \big\} \\
& + \dot{\tilde{\beta}}^s I^s - I^s \dot{\tilde{\beta}}^s \Big] \{\mathbf{b}^k\} \cdot \boldsymbol{\omega}^k \\
& - \sum_{s=1}^{n_k} (\mathbf{R}^k + \mathbf{r}^s) \times m_s \mathring{\mathring{\mathbf{u}}}^s - \boldsymbol{\omega}^k \times \sum_{s=1}^{n_k} (\mathbf{R}^k + \mathbf{r}^s) \times m_s \mathring{\mathbf{u}}^s \\
& - \sum_{s=1}^{n_k} (\mathbf{I}^s \cdot \mathring{\mathring{\boldsymbol{\beta}}}^s + \boldsymbol{\omega}^k \times \mathbf{I}^s \cdot \mathring{\boldsymbol{\beta}}^s)
\end{aligned}
\tag{28}
$$

Eliminating the use of $\mathbf{R}^k$ for simplicity (noting that this is an arbitrary vector fixed in $\mathscr{b}_k$ and it can always be chosen as zero) and substituting q^k and q^r where appropriate, the matrix form of Eq. (28) becomes

$$
\begin{aligned}
W^k = {} & T^k + \sum_{r\in\mathfrak{B}} \tilde{D}^{kr} C^{kr} F^r + \left[\tilde{F}^k - \left(C^{k0} \frac{\mathfrak{M}_k}{\mathfrak{M}} F \right)^{\sim} \right] c^k \\
& + \sum_{r\in\mathfrak{B}} \mathfrak{M}_r \tilde{D}^{kr} C^{kr} \left[-\Sigma_{U0}^T \frac{M^r}{\mathfrak{M}_r} \ddot{q}^r + 2\tilde{\omega}^r \dot{c}^r - \tilde{c}^r \dot{\omega}^r + \tilde{\omega}^r \tilde{\omega}^r c^r \right] \\
& + \mathfrak{M}_k \tilde{c}^k \sum_{r\in\mathfrak{B}} C^{kr} [-\tilde{D}^{rk} \dot{\omega}^r + \tilde{\omega}^r \tilde{\omega}^r D^{rk}] \\
& - \sum_{r\in\mathfrak{B}} \Phi^{kr} C^{kr} \dot{\omega}^r + \mathfrak{M} \sum_{r\in\mathfrak{B}-k} \tilde{D}^{kr} C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk} - \tilde{\omega}^k \Phi^{kk} \omega^k \\
& - \dot{h}^k - \tilde{\omega}^k h^k - \Big[2(M^k r_k)^T \dot{q}^k U - r_k^\dagger (M^k \dot{q}^k)^{\dagger T} - (M^k \dot{q}^k)^\dagger r_k^{\dagger T} \\
& + \Sigma_{0U}^T (\dot{\tilde{q}}^k M^k - M^k \dot{\tilde{q}}^k) \Sigma_{0U} \Big] \omega^k - \Sigma_{U0}^T \tilde{r}_k M^k \ddot{q}^k - \tilde{\omega}^k \Sigma_{U0}^T \tilde{r}_k M^k \dot{q}^k \\
& - \Sigma_{0U}^T M^k \ddot{q}^k - \tilde{\omega}^k \Sigma_{0U}^T M^k \dot{q}^k
\end{aligned}
\tag{29}
$$

where the operator † reassembles the 3 by 1 submatrices of a column matrix into a three-row matrix, as illustrated by

$$r_k^{\dagger} \equiv [r^1 \quad 0 \quad r^2 \quad 0 \quad \cdots \quad r^{n_k} \quad 0]$$

Using the identity

$$(M^k \dot{q}^k)^T r_k \equiv (M^k r_k)^T \dot{q}^k$$

and regrouping some of the terms in (29), we have

$$\begin{aligned}
W^k = &- \sum_{r \in \mathfrak{B}} \left[\Phi^{kr} C^{kr} + \mathfrak{M}_k \tilde{c}^k C^{kr} \tilde{D}^{rk} + \mathfrak{M}_r \tilde{D}^{kr} C^{kr} \tilde{c}^r \right] \dot{\omega}^r \\
&- [\Sigma_{0U}^T + \Sigma_{U0}^T \tilde{r}_k] M^k \ddot{q}^k - \sum_{r \in \mathfrak{B}} \tilde{D}^{kr} C^{kr} \Sigma_{U0}^T M^r \ddot{q}^r - \dot{h}^k \\
&+ T^k + \sum_{r \in \mathfrak{B}} \tilde{D}^{kr} C^{kr} F^r + \left[\tilde{F}^k - \left(C^{k0} \frac{\mathfrak{M}_k}{\mathfrak{M}} F \right)^{\sim} \right] c^k \\
&+ \sum_{r \in \mathfrak{B}} \mathfrak{M}_r \tilde{D}^{kr} C^{kr} (2 \tilde{\omega}^r \dot{c}^r + \tilde{\omega}^r \tilde{\omega}^r c^r) + \mathfrak{M}_k \tilde{c}^k \sum_{r \in \mathfrak{B}} C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk} \\
&+ \mathfrak{M} \sum_{r \in \mathfrak{B} - k} \tilde{D}^{kr} C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk} - \tilde{\omega}^k \Phi^{kk} \omega^k - \tilde{\omega}^k h^k \\
&- \Big[2 (M^k \dot{q}^k)^T r_k U - r_k^{\dagger} (M^k \dot{q}^k)^{\dagger T} - (M^k \dot{q}^k)^{\dagger} r_k^{\dagger T} \\
&+ \Sigma_{0U}^T (\tilde{\dot{q}}^k M^k - M^k \tilde{\dot{q}}^k) \Sigma_{0U} \Big] \omega^k - \tilde{\omega}^k (\Sigma_{0U}^T + \Sigma_{U0}^T \tilde{r}_k) M^k \dot{q}^k
\end{aligned} \tag{30}$$

The truncated modal coordinates, $\bar{\delta}^k$ and $\bar{\eta}^k$, may now be introduced into the kth substructure equation by way of Eq. (27), as follows:

$$\begin{aligned}
W^k = &- \sum_{r \in \mathfrak{B}} \left[\Phi^{kr} C^{kr} + \mathfrak{M}_k \tilde{c}^k C^{kr} \tilde{D}^{rk} + \mathfrak{M}_r D^{kr} C^{kr} \tilde{c}^r \right] \dot{\omega}^r \\
&- \bar{\Delta}_R^{k^T} \ddot{\bar{\delta}}^k - \bar{\Delta}_I^{k^T} \ddot{\bar{\eta}}^k + \sum_{r \in \mathfrak{B}} \tilde{D}^{kr} C^{kr} \left[G_r \bar{\sigma}^r \ddot{\bar{\delta}}^r + P_r \bar{\sigma}^r \ddot{\bar{\eta}}^r \right] \\
&- \dot{h}^k + T^k + \sum_{r \in \mathfrak{B}} \tilde{D}^{kr} C^{kr} F^r + \left[\tilde{F}^k - \left(C^{k0} \frac{\mathfrak{M}_k}{\mathfrak{M}} F \right)^{\sim} \right] c^k \\
&+ \sum_{r \in \mathfrak{B}} \left[\mathfrak{M}_r \tilde{D}^{kr} C^{kr} (2 \tilde{\omega}^r \dot{c}^r + \tilde{\omega}^r \tilde{\omega}^r c^r) + \mathfrak{M}_k \tilde{c}^k C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk} \right] \\
&+ \mathfrak{M} \sum_{r \in \mathfrak{B} - k} \tilde{D}^{kr} C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk} - \tilde{\omega}^k \Phi^{kk} \omega^k - \tilde{\omega}^k h^k \\
&- j^k \omega^k - \tilde{\omega}^k \left(\bar{\Delta}_R^{k^T} \dot{\bar{\delta}}^k + \bar{\Delta}_I^{k^T} \dot{\bar{\eta}}^k \right)
\end{aligned} \tag{31}$$

where

$$\bar{\Delta}_R^k = -2\bar{\sigma}^k \bar{\Gamma}_k^T M^k [\Sigma_{0U} - \tilde{r}_k \Sigma_{U0}]$$

$$\Delta_I^k = -2\bar{\sigma}^k \bar{\psi}_k^T M^k [\Sigma_{0U} - \tilde{r}_k \Sigma_{U0}]$$

$$\bar{P}_k = 2\Sigma_{U0}^T M^k \bar{\psi}_k$$

$$\bar{G}_k = 2\Sigma_{U0}^T M^k \bar{\Gamma}_k$$

$$j^k = 2(M^k \dot{q}^k)^T r_k - r_k^\dagger (M^k \dot{q}^k)^{\dagger T} - (M^k \dot{q}^k)^\dagger r_k^{\dagger T} + \Sigma_{0\dot{U}}^T (\tilde{\dot{q}}^k M^k - M^k \tilde{\dot{q}}^k) \Sigma_{0\dot{U}}$$

$$\Phi^{kk} = K^k + J^k$$

$$\dot{\Phi}^{kk} = j^k$$

Using the relation in Eq. (6), the vehicle equations, (2) and (3), become (in matrix form)

$$A^{00}\dot{\omega}^0 + \sum_{j \in \mathcal{P}} A^{0j}\ddot{\gamma}_j + \sum_{m \in \mathcal{F}} A_R^{0m} \ddot{\delta}^m + \sum_{m \in \mathcal{F}} A_I^{0m} \ddot{\eta}^m = \sum_{k \in \mathcal{B}} C^{0k} E^k \tag{32}$$

and for $s \in \mathcal{P}$,

$$A^{s0}\dot{\omega}^0 + \sum_{j \in \mathcal{P}} A^{sj}\ddot{\gamma}_j + \sum_{m \in \mathcal{F}} A_R^{sm} \ddot{\delta}^m + \sum_{m \in \mathcal{F}} A_I^{sm} \ddot{\eta}^m = g^{s^T} \sum_{k \in \mathcal{P}} \epsilon_{sk} C^{sk} E^k + \tau_s \tag{33}$$

where

$$A^{00} = \sum_{k \in \mathcal{B}} \sum_{r \in \mathcal{B}} C^{0k}\left(\Phi^{kr}C^{kr} + \mathfrak{M}_k \tilde{c}^k C^{kr} \tilde{D}^{rk} + \mathfrak{M}_r \tilde{D}^{kr} C^{kr} \tilde{c}^r\right) C^{r0}, \quad 3 \text{ by } 3 \tag{34}$$

$$A^{0j} = \sum_{k \in \mathcal{B}} \sum_{r \in \mathcal{P}} C^{0k}\left(\Phi^{kr}C^{kr} + \mathfrak{M}_k \tilde{c}^k C^{kr} \tilde{D}^{rk} + \mathfrak{M}_r \tilde{D}^{kr} C^{kr} \tilde{c}^r\right) \epsilon_{jr} C^{rj} g^j, \quad 3 \text{ by } 1 \tag{35}$$

$$A^{s0} = g^{s^T} \sum_{k \in \mathcal{P}} \sum_{r \in \mathcal{B}} \epsilon_{sk} C^{sk}\left(\Phi^{kr}C^{kr} + \mathfrak{M}_k \tilde{c}^k C^{kr} \tilde{D}^{rk} + \mathfrak{M}_r \tilde{D}^{kr} C^{kr} \tilde{c}^r\right) C^{r0}, \quad 1 \text{ by } 3 \tag{36}$$

$$A^{sj} = g^{s^T} \sum_{k \in \mathcal{P}} \sum_{r \in \mathcal{B}} \epsilon_{sk}\, \epsilon_{jr} C^{sk}\left(\Phi^{kr}C^{kr} + \mathfrak{M}_k \tilde{c}^k C^{kr} \tilde{D}^{rk} + \mathfrak{M}_r \tilde{D}^{kr} C^{kr} \tilde{c}^r\right) C^{rj} g^j, \quad 1 \text{ by } 1 \tag{37}$$

$$A_R^{0m} = C^{0m} \bar{\Delta}_R^{m^T} - \sum_{r \in \mathcal{B}} C^{0r} \tilde{D}^{rm} C^{rm} \bar{G}_m \bar{\sigma}^m, \quad 3 \text{ by } N_m \tag{38}$$

$$A_I^{0m} = C^{0m} \bar{\Delta}_I^{m^T} - \sum_{r \in \mathcal{B}} C^{0r} \tilde{D}^{rm} C^{rm} \bar{P}_m \bar{\sigma}^m, \quad 3 \text{ by } N_m \tag{39}$$

$$A_R^{sm} = g^{s^T}\left(\epsilon_{sm} C^{sm} \bar{\Delta}_R^{m^T} - \sum_{r \in \mathfrak{B}} \epsilon_{sr} C^{sr} \tilde{D}^{rm} C^{rm} \bar{G}_m \bar{\sigma}^m\right), \quad 1 \text{ by } N_m \tag{40}$$

$$A_I^{sm} = g^{s^T}\left(\epsilon_{sm} C^{sm} \bar{\Delta}_I^{m^T} - \sum_{r \in \mathfrak{B}} \epsilon_{sr} C^{sr} \tilde{D}^{rm} C^{rm} \bar{P}_m \bar{\sigma}^m\right), \quad 1 \text{ by } N_m \tag{41}$$

$\mathfrak{F} \equiv$ the integer set containing the labels of only those rigid bodies of the system that possess a nonrigid appendage.

$$E^k = T^k - \tau_R^k - \tilde{\omega}^k \mathcal{J}^k(\tilde{\omega}^k + \dot{\psi}_R^k) + \sum_{r \in \mathfrak{B}} \tilde{D}^{kr} C^{kr} F^r$$

$$+ \left[\tilde{F}^k - \left(C^{k0} \frac{\mathfrak{M}_k}{\mathfrak{M}} F\right)^{\sim}\right] c^k + \mathfrak{M} \sum_{r \in \mathfrak{B} - k} \tilde{D}^{kr} C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk}$$

$$- \tilde{\omega}^k \Phi^{kk} \omega^k - \dot{j}^k \omega^k - \tilde{\omega}^k \left(\bar{\Delta}_R^{k^T} \dot{\bar{\delta}}^k + \bar{\Delta}_I^{k^T} \dot{\bar{\eta}}^k\right)$$

$$- \sum_{r \in \mathfrak{B}} \left[\left(\Phi^{kr} C^{kr} + \mathfrak{M}_k \tilde{c}^k C^{kr} \tilde{D}^{rk} + \mathfrak{M}_r \tilde{D}^{kr} C^{kr} \tilde{c}^r\right) \cdot \sum_{j \in \mathfrak{P}} \epsilon_{jr} C^{rj} \tilde{\omega}^j g^j \dot{\gamma}_j\right.$$

$$\left. - \mathfrak{M}_r \tilde{D}^{kr} C^{kr} (2\tilde{\omega}^r \dot{\tilde{c}}^r + \tilde{\omega}^r \tilde{\omega}^r c^r) - \mathfrak{M}_k \tilde{c}^k C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk}\right], \quad 3 \text{ by } 1 \tag{42}$$

and substitutions have been made for $\dot{h}^k$ and h^k based on restriction to three orthogonal axisymmetric rotors in $\mathscr{b}_k$, with spin axes aligned to the unit vectors $\{\mathbf{b}^k\}$, and the following equations:

$$h^k = \mathcal{J}^k \dot{\psi}_R^k \tag{43}$$

$$\tau_R^k = \mathcal{J}^k (\ddot{\psi}_R^k + \dot{\omega}^k) \tag{44}$$

$$\therefore \quad \dot{h}^k = \tau_R^k - \mathcal{J}^k \dot{\omega}^k \tag{45}$$

where

$\dot{\psi}_R^k \equiv \dot{\boldsymbol{\psi}}_R^k \cdot \{\mathbf{b}^k\} =$ 3 by 1 matrix of components of spin rate relative to $\mathscr{b}_k$ for three orthogonal axisymmetric rotors in $\mathscr{b}_k$.

$\mathcal{J}^k \equiv$ *spin-axis* inertia matrix (diagonal) for the three axisymmetric rotors in $\mathscr{b}_k$.

$\tau_R^k \equiv \boldsymbol{\tau}_R^k \cdot \{\mathbf{b}^k\} =$ 3 by 1 matrix of applied torque on the three axisymmetric rotors in $\mathscr{b}_k$.

It is to be understood that when symmetric rotors are present in the kth substructure, the rotors' mass and moments of inertia are to be included in $\mathbf{J}^k$, the undeformed substructure's inertia dyadic for o_k. Of course, the mass of the rotors is also to be included in the substructure mass and c.m.-location calculations.

Equation (44) then provides up to three scalar differential equations which are uncoupled in acceleration from the system's vehicle/appendage equations. They may be integrated and, with ω^k and τ_R^k known, can be solved for $\dot{\psi}_R^k$, which is then supplied to Eq. (42).

If one now operates on the appendage equations, Eqs. (26), in a similar way, they may be expressed as

$m \in \mathcal{F}$:

$$\mathcal{Q}^{m0}\dot{\omega}^0 + \sum_{j\in\mathcal{P}} \mathcal{Q}^{mj}\ddot{\gamma}_j + \sum_{n\in\mathcal{F}} \mathcal{Q}_R^{mn}\dot{\bar{\delta}}^n + \sum_{n\in\mathcal{F}} \mathcal{Q}_I^{mn}\dot{\bar{\eta}}^n = Q_R^m \tag{46}$$

$$\mathcal{D}^{m0}\dot{\omega}^0 + \sum_{j\in\mathcal{P}} \mathcal{D}^{mj}\ddot{\gamma}_j + \sum_{n\in\mathcal{F}} \mathcal{D}_R^{mn}\dot{\bar{\delta}}^n + \sum_{n\in\mathcal{F}} \mathcal{D}_I^{mn}\dot{\bar{\eta}}^n = Q_I^m \tag{47}$$

where

$$\mathcal{Q}^{m0} = \frac{1}{2}\left[\bar{\Delta}_R^m C^{m0} + \bar{\sigma}^m \bar{G}_m^T \sum_{r\in\mathcal{B}} C^{mr}\tilde{D}^{rm}C^{r0}\right], \quad N_m \text{ by } 3 \tag{48}$$

$$\mathcal{Q}^{mj} = \frac{1}{2}\left[\bar{\Delta}_R^m \epsilon_{jm} C^{mj} + \bar{\sigma}^m \bar{G}_m^T \sum_{r\in\mathcal{B}} C^{mr}\tilde{D}^{rm}\epsilon_{jr}C^{rj}\right] g^j, \quad N_m \text{ by } 1 \tag{49}$$

$$\mathcal{Q}_R^{mn} = -\frac{1}{2}\bar{\sigma}^m \bar{G}_m^T C^{mn}\bar{G}_n \frac{\bar{\sigma}^n}{\mathfrak{M}}, \quad (m \neq n); \quad N_m \text{ by } N_n \tag{50}$$

$$\mathcal{Q}_R^{mn} = U, \quad (m = n); \quad N_m \text{ by } N_m$$

$$\mathcal{Q}_I^{mn} = -\frac{1}{2}\bar{\sigma}^m \bar{G}_m^T C^{mn}\bar{P}_n \frac{\bar{\sigma}^n}{\mathfrak{M}}, \quad (m \neq n); \quad N_m \text{ by } N_n \tag{51}$$

$$\mathcal{Q}_I^{mn} = 0, \quad (m = n); \quad N_m \text{ by } N_m$$

$$\mathcal{D}^{m0} = \frac{1}{2}\left[\bar{\Delta}_I^m C^{m0} + \bar{\sigma}^m \bar{P}_m^T \sum_{r\in\mathcal{B}} C^{mr}\tilde{D}^{rm}C^{r0}\right], \quad N_m \text{ by } 3 \tag{52}$$

$$\mathcal{D}^{mj} = \frac{1}{2}\left[\bar{\Delta}_I^m \epsilon_{jm} C^{mj} + \bar{\sigma}^m \bar{P}_m^T \sum_{r\in\mathcal{B}} C^{mr}\tilde{D}^{rm}\epsilon_{jr}C^{rj}\right] g^j, \quad N_m \text{ by } 1 \tag{53}$$

$$\mathcal{D}_R^{mn} = -\frac{1}{2}\bar{\sigma}^m \bar{P}_m^T C^{mn}\bar{G}_n \frac{\bar{\sigma}^n}{\mathfrak{M}}, \quad (m \neq n); \quad N_m \text{ by } N_n \tag{54}$$

$$\mathcal{D}_R^{mn} = 0, \quad (m = n); \quad N_m \text{ by } N_m \tag{55}$$

$$\mathcal{D}_I^{mn} = -\frac{1}{2}\bar{\sigma}^m \bar{P}_m^T C^{mn}\bar{P}_n \frac{\bar{\sigma}^n}{\mathfrak{M}}, \quad (m \neq n); \quad N_m \text{ by } N_n \tag{56}$$

$$\mathcal{D}_I^{mn} = U, \quad (m = n); \quad N_m \text{ by } N_m \tag{57}$$

$$Q_R^m = +\bar{\sigma}^m\left[-\bar{\eta}^m - \bar{\xi}^m\bar{\delta}^m + \frac{1}{2}\,\bar{G}_m^T V_m - \bar{\Gamma}_m^T X_m\right] - Z_R^m, \quad N_m \text{ by } 1 \tag{58}$$

$$Q_I^m = +\bar{\sigma}^m\left[\bar{\delta}^m - \bar{\xi}^m\bar{\eta}^m + \frac{1}{2}\,\bar{P}_m^T V_m - \bar{\psi}_m^T X_m\right] - Z_I^m, \quad N_m \text{ by } 1 \tag{59}$$

$$V_m = C^{m0}\,\frac{F}{\mathfrak{M}} + \sum_{r\in\mathfrak{B}} C^{mr}\tilde{\omega}^r\tilde{\omega}^r D^{rm}$$

$$+ \sum_{r\in\mathfrak{B}-m} C^{mr}\,\frac{\mathfrak{M}_r}{\mathfrak{M}}\left(2\tilde{\omega}^r\dot{c}^r + \tilde{\omega}^r\tilde{\omega}^r c^r\right) - \tilde{\Omega}^m\tilde{\Omega}^m D^{mm}, \quad 3 \text{ by } 1 \tag{60a}$$

$$X_m = \lambda^m - M^m(\Sigma_{U0}\omega^m)^{\sim}(\Sigma_{U0}\omega^m)^{\sim} r_m - (\Sigma_{0U}\omega^m)^{\sim} M^m(\Sigma_{0U}\omega^m)$$

$$+ M^m(\Sigma_{U0}\Omega^m)^{\sim}(\Sigma_{U0}\Omega^m)^{\sim} r_m + (\Sigma_{0U}\Omega^m)^{\sim} M^m(\Sigma_{0U}\Omega^m), \quad n_m \text{ by } 1 \tag{60b}$$

$$Z_R^m = \frac{1}{2}\sum_{j\in\mathcal{P}}\left(\bar{\Delta}_R^m\epsilon_{jm}C^{mj} + \bar{\sigma}^m\bar{G}_m^T\sum_{r\in\mathfrak{B}} C^{mr}\tilde{D}^{rm}\epsilon_{jr}C^{rj}\right)\tilde{\omega}^j g^j\dot{\gamma}_j, \quad N_m \text{ by } 1 \tag{61a}$$

$$Z_I^m = \frac{1}{2}\sum_{j\in\mathcal{P}}\left(\bar{\Delta}_I^m\epsilon_{jm}C^{mj} + \bar{\sigma}^m\bar{P}_m^T\sum_{r\in\mathfrak{B}} C^{mr}\tilde{D}^{rm}\epsilon_{jr}C^{rj}\right)\tilde{\omega}^j g^j\dot{\gamma}_j, \quad N_m \text{ by } 1 \tag{61b}$$

Recapping, the system equations (minus the rotor equations) are as follows:

$$A^{00}\dot{\omega}^0 + \sum_{j\in\mathcal{P}} A^{0j}\ddot{\gamma}_j + \sum_{m\in\mathcal{F}} A_R^{0m}\ddot{\bar{\delta}}^m + \sum_{m\in\mathcal{F}} A_I^{0m}\ddot{\bar{\eta}}^m = \sum_{k\in\mathfrak{B}} C^{0k}E^k \tag{62a}$$

$s \in \mathcal{P}$:

$$A^{s0}\dot{\omega}^0 + \sum_{j\in\mathcal{P}} A^{sj}\ddot{\gamma}_j + \sum_{m\in\mathcal{F}} A_R^{sm}\ddot{\bar{\delta}}^m + \sum_{m\in\mathcal{F}} A_I^{sm}\ddot{\bar{\eta}}^m = g^{s^T}\sum_{k\in\mathcal{P}}\epsilon_{sk}C^{sk}E^k + \tau_s \tag{62b}$$

$m \in \mathcal{F}$:

$$\mathcal{A}^{m0}\dot{\omega}^0 + \sum_{j\in\mathcal{P}} \mathcal{A}^{mj}\ddot{\gamma}_j + \sum_{n\in\mathcal{F}} \mathcal{A}_R^{mn}\ddot{\bar{\delta}}^n + \sum_{n\in\mathcal{F}} \mathcal{A}_I^{mn}\ddot{\bar{\eta}}^n = Q_R^m \tag{62c}$$

$m \in \mathcal{F}$:

$$\mathcal{D}^{m0}\dot{\omega}^0 + \sum_{j\in\mathcal{P}} \mathcal{D}^{mj}\ddot{\gamma}_j + \sum_{n\in\mathcal{F}} \mathcal{D}_R^{mn}\ddot{\bar{\delta}}^n + \sum_{n\in\mathcal{F}} \mathcal{D}_I^{mn}\ddot{\bar{\eta}}^n = Q_I^m \tag{62d}$$

and these may be combined into the single matrix equation of the form $A\dot{x} = B$, as shown in Eq. (63).

$$
\left[\begin{array}{c|c|c|c}
\underset{(3\times 3)}{A^{00}} & \underset{(3\times n_n)}{A^{0j}} & \underset{(3\times N_m)}{\sum_{m\in\mathcal{F}} A_R^{0m}} & \underset{(3\times N_m)}{\sum_{m\in\mathcal{F}} A_I^{0m}} \\
\hline
\underset{(n_n\times 3)}{A^{s0}} & \underset{(n_n\times n_n)}{A^{sj}} & \underset{(n_n\times N_m)}{\sum_m A_R^{sm}} & \underset{(n_n\times N_m)}{\sum_m A_I^{sm}} \\
\hline
\underset{(N_m\times 3)}{\mathcal{C}^{m0}} & \underset{(N_m\times n_n)}{\mathcal{C}^{mj}} & \underset{(N_m\times N_n)}{\sum_n \mathcal{C}_R^{mn}} & \underset{(N_m\times N_n)}{\sum_n \mathcal{C}_I^{mn}} \\
\hline
\underset{(N_m\times 3)}{\mathcal{D}^{m0}} & \underset{(N_m\times n_n)}{\mathcal{D}^{mj}} & \underset{(N_m\times N_n)}{\sum_n \mathcal{D}_R^{mn}} & \underset{(N_m\times N_n)}{\sum_n \mathcal{D}_I^{mn}}
\end{array}\right]
\left[\begin{array}{c}
\underset{(3\times 1)}{\dot{\omega}^0} \\
\hline
\underset{(n_n\times 1)}{\ddot{\gamma}_j} \\
\hline
\underset{(N_m\times 1)}{\ddot{\delta}^m} \\
\hline
\underset{(N_m\times 1)}{\ddot{\eta}^m}
\end{array}\right]
=
\left[\begin{array}{c}
\underset{(3\times 1)}{\sum_{k\in\mathcal{B}} C^{0k}E^k} \\
\hline
\underset{(n_n\times 1)}{g^{s^T}\sum_{k\in\mathcal{P}} \epsilon_{sk} C^{sk}E^k + \tau_s} \\
\hline
\underset{(N_m\times 1)}{Q_R^m} \\
\hline
\underset{(N_m\times 1)}{Q_I^m}
\end{array}\right]
\qquad (63)
$$

Except for $\mathcal{C}_R^{mn}$, $\mathcal{C}_I^{mn}$, $\mathcal{D}_R^{mn}$, and $\mathcal{D}_R^{mn}$ when $m = n$, the elements of system matrix A are, in general, time-variable. Note also that, if the appendage equations are multiplied through by the factor 2, matrix A becomes symmetric.

B. Subroutine MBDYFR

Equation (63) provides a complete set of rotational dynamics equations which lend themselves to solution by means of a generic computer program or subroutine for the rotating appendage case. When augmented by the rotor equations, control equations, and kinematical equations, they are fully descriptive of the system behavior.

The kinematical variables adopted in the preceding sections are as follows: γ_k for $k \in \mathcal{P}$ (Def. 23); C^{rj} for $r, j \in \mathcal{B}$ (Def. 31); and $\omega^0 \equiv \{\mathbf{b}^0\} \cdot \boldsymbol{\omega}^0$ (Def. 9). Although the equations of motion have been expressed in terms of these quantities, the latter are not all independent. Relationships among kinematical variables developed in this section must therefore either be considered in conjunction with the dynamical equations or be substituted into them to remove redundant variables whenever a solution is sought.

The direction cosine matrix C^{rj} (Def. 31) relates sets of orthogonal unit vectors fixed in $\mathscr{b}_r$ and $\mathscr{b}_j$, and hence depends upon those angles γ_α for which $\mathscr{h}_\alpha$ lies between $\mathscr{b}_r$ and $\mathscr{b}_j$, and also upon the corresponding unit vectors $\mathbf{g}^\alpha$ defining the intervening hinge axes. For the special case in which $\mathscr{b}_r$ and $\mathscr{b}_j$ are *contiguous* and $j < r$, it is always possible to express C^{rj} (and C^{jr}) in terms of the single angle γ_r and the single matrix g^r, as follows:

$$C^{rj} = U \cos \gamma_r - \tilde{g}^r \sin \gamma_r + \tilde{g}^r g^{r^T}(1 - \cos \gamma_r)$$

and

$$C^{jr} = U \cos \gamma_r + \tilde{g}^r \sin \gamma_r + \tilde{g}^r g^{r^T}(1 - \cos \gamma_r) = (C^{rj})^T$$

It is only required that C^{rj} be determined where $\mathscr{b}_r$ and $\mathscr{b}_j$ are contiguous and, since $C^{0r} = C^{0j}C^{jr}$, to then derive matrices C^{0r} for $r \in \mathscr{P}$. An algorithm for accomplishing this task is described in Ref. 6, Appendix A.

The Fortran V subroutine, called MBDYFR, which provides the solution to Eq. (63), has been designed in much the same form as those subroutines described in Refs. 6 and 7. The routine may be exercised by means of either of two call statements. An initializing call statement supplies the routine with data that will remain constant throughout the dynamic simulation run.

The description which follows of the subroutine initialization statement includes some new variables which will now be defined. The use of these new variables is necessitated by the desire to make the subroutine MBDYFR more efficient. Therefore, the convention (described in Defs. 1–4) of labeling the system's rigid bodies from 0 to n, where each connection between bodies is a line hinge, will be modified. Rather than introduce imaginary massless bodies at connections with 2 or 3 degrees of rotational freedom, these types of connections will be handled directly by the routine and no new bodies will be introduced.

Def. 45. Let n_c be the number of *connections* joining a set of $n_c + 1$ substructures. A connection is a 1-, 2-, or 3-degree-of-freedom joint at which all the rotational axes share a common point. The axes need not be mutually orthogonal.

Def. 46. Define the integer set $\mathscr{B}^r \equiv \{0, 1, \ldots, n_c\}$.

Def. 47. Define the integer set $\mathscr{P}^r \equiv \{1, 2, \ldots, n_c\}$.

Def. 48. Let $\mathscr{B}^r_j$ be the jth neighbor set for $j \in \mathscr{B}^r$, such that $k \in \mathscr{B}^r_j$ if $\mathscr{b}_k$ is attached to $\mathscr{b}_j$.

The rigid body labeling process is to be carried out precisely as prescribed in Def. 4, except that the last label will be $\mathscr{b}_{n_c}$ rather than $\mathscr{b}_n$. Note, however, that the connecting joint degrees of freedom are still labeled from 1 to n, so that one still has $\gamma_1, \gamma_2, \ldots, \gamma_n$ and $\mathbf{g}^1, \mathbf{g}^2, \ldots, \mathbf{g}^n$ (The joints *must* be in the sequence corresponding to the body label sequence, as shown in Fig. 2). All references to the "kth substructure," when applying the MBDYFR subroutine, imply that $k \in \mathscr{B}^r$.

Def. 49. For $k \in \mathscr{P}^r$, let $\mathscr{k}_k$ denote the index label of the body attached to $\mathscr{b}_k$ and on the path leading to $\mathscr{b}_0$. The scalars $\mathscr{k}_k$ will be termed "connection elements." Thus, it is always true that $\mathscr{k}_1 = 0$.

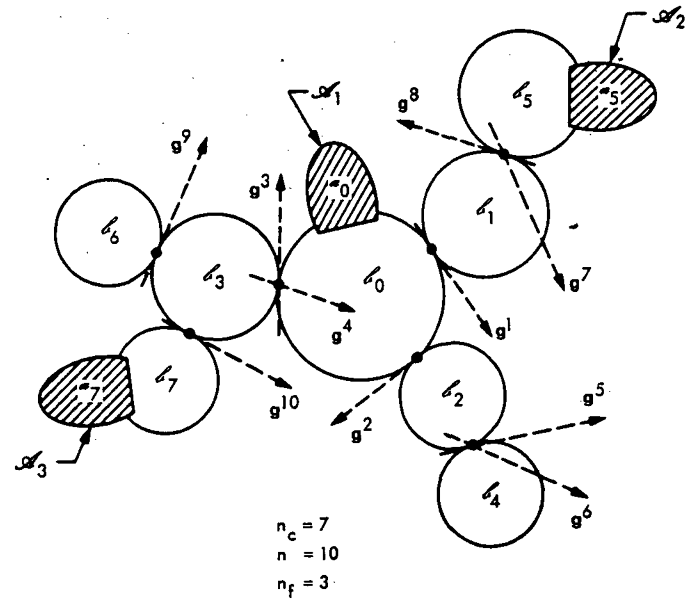

Fig. 2. An 8-body, 10-hinge system illustrating the labeling convention

Def. 50. Let d_k, $k \in \mathcal{P}^r$, denote the number of degrees of freedom at the kth connection.

It is also necessary, when applying the subroutine, to relabel each of the nonrigid appendages α_k *in the same sequence* from 1 to n_f (see Fig. 2) so that the labels become $\mathcal{Q}_1, \ldots, \mathcal{Q}_{n_f}$.

Def. 51. Let n_f be the number of nonrigid appendages in the system (no more than one per substructure).

The first column of the input array, F, contains the index labels of those rigid bodies to which nonrigid appendages $\mathcal{Q}_i$ $(i = 1, \ldots, n_f)$ are attached.

Initializing Call Statement

CALL MBDYFR(NC, H, MB, MS, PB, PS, G, PI,
NF, F, ER, EI, SR, MF, RF, WF, ZF)

where

NC = the integer n_c = number of system *connections* (see Def. 45).

H(k, m) = array containing the connection elements h_k, $k \in \mathcal{P}^r$, and the number of degrees of freedom, d_k, at the connection; $m = 1, 2$. (H(1, 1) = h_1, H(2, 1) = h_2, . . . , H(n_c, 1) = h_{n_c}, H(1, 2) = d_1, H(2, 2) = d_2, . . . , H(n_c, 2) = d_{n_c}.)

MB(j) = array of undeformed reference substructure (b_0) inertial constants $j = 1, \ldots, 7$. (Specifically: MB(1) = $\bar{J}^0_{11}$, MB(2) = $\bar{J}^0_{22}$, MB(3) = $\bar{J}^0_{33}$, MB(4) = $-\bar{J}^0_{12}$, MB(5) = $-\bar{J}^0_{13}$, MB(6) = $-\bar{J}^0_{23}$, MB(7) = $\mathcal{M}_0$.)

MS(i, j) = array of remaining substructure body (undeformed) inertial constants; $i \in \mathcal{P}^r$; $j = 1, \ldots, 7$. (Thus: MS(i, 1) $= \bar{J}^i_{11}$, MS(i, 2) $= \bar{J}^i_{22}, \ldots$, MS(i, 7) $= \mathfrak{M}_i$.

PB(i, j) = array containing elements of p^{0i}; $i \in \mathcal{B}^r_0$, $j \in 1, 2, 3$.

PS(i, j, k) = array containing elements of p^{ij}; $i \in \mathcal{P}^r$, $j \in \mathcal{B}^r_i$, $k = 1, 2, 3$. (Exception!! If $j < i$, set PS(i, i, k) $= p^{ij}$. Example: PS(3, 3, 1) $= p^{32}_1$. All PS(i, j, k), where $j < i$, will be ignored.)

G(i, j) = array containing elements of g^i; $i \in \mathcal{P}$, $j = 1, 2, 3$.

PI(i) = array of indicators; $i = 1, 2, \ldots, n + 1$. (If γ_i is a *prescribed* variable, PI(i) = 1. Otherwise, PI(i) = 0. Also, if PI($n + 1$) = 1, system angular momentum HM will be calculated; otherwise, HM is set to zero.

NF = the integer n_f = number of substructures with nonrigid appendages = number of nonrigid appendages.

F(n, m) = array containing the index labels of those rigid bodies with nonrigid appendages, the number of nodal bodies in each appendage's finite element model, and the number of modes to be used in each appendage's modal model; $n = 1, 2, \ldots, n_f$, $m = 1, 2, 3$. (Thus:

F(1, 1) = index label of rigid body carrying appendage $\mathcal{Q}_1$

F(1, 2) = number of nodal bodies in appendage $\mathcal{Q}_1$

F(1, 3) = number of modes representing appendage $\mathcal{Q}_1$

F(2, 1) = index label of rigid body carrying appendage $\mathcal{Q}_2$

⋮

etc.

⋮

F(n_f, 3) = number of modes representing appendage $\mathcal{Q}_{n_f}$.)

ER(n, i, j) = array of elements of $\bar{\psi}^j_k$; $n = 1, 2, \ldots, n_f$; $i = 1, 2, \ldots, 6n_k$; k = F(n, 1); $j = 1, 2, \ldots, N_k$.

EI(n, i, j) = array of elements of $\bar{\Gamma}^j_k$; $n = 1, 2, \ldots, n_f$; $i = 1, 2, \ldots, 6n_k$; k = F(n, 1); $j = 1, 2, \ldots, N_k$.

SR(n, j) = array of substructure nominal spin rates, Ω^k, k = F(n, 1); $n = 1, 2, \ldots, n_f$; $j = 1, 2, 3$.

MF(n, i, j) = array of nodal body inertial properties, M^k, for each nonrigid appendage; $n = 1, 2, \ldots, n_f$; $i = 1, 2, \ldots, n_k$; k = F(n, 1); $j = 1, 2, \ldots, 7$. (Example: MF(2, 3, 1) $= I^3_{11}$, MF(2, 3, 2) $= I^3_{22}$, MF(2, 3, 3) $= I^3_{33}$, MF(2, 3, 4) $= -I^3_{12}, \ldots$, MF(2, 3, 7) $= m_3$, all for nonrigid appendage $\mathcal{Q}_2$, third nodal body.)

RF(n, i, j) = array of elements of r_k, $k = F(n, 1)$, for each nonrigid appendage; $n = 1, 2, \ldots, n_f$, $i = 1, 2, \ldots, n_k$, $j = 1, 2, 3$. (Example: RF(1, 5, 1) = r_1^5, RF(1, 5, 2) = r_2^5, RF(1, 5, 3) = r_3^5, all for appendage $\mathcal{A}_1$.)

WF(n, j) = array of modal frequencies, $\bar{\sigma}^k$, $k = F(n, 1)$, for each nonrigid appendage; $n = 1, 2, \ldots, n_f, j = 1, 2, \ldots, N_k$.

ZF(n, j) = array of modal damping factors, $\bar{\xi}^k$, $k = F(n, 1)$, for each nonrigid appendage; $n = 1, 2, \ldots, n_f, j = 1, 2, \ldots, N_k$.

The statement CALL MBDYFR (NC, H, . . .) need only be executed *once* prior to a simulation run. However, as the simulation proceeds, the routine must be entered at every numerical integration step to compute the angular accelerations $\dot{\omega}^0, \ddot{\gamma}_1, \ldots, \ddot{\gamma}_n$ and the modal coordinate acceleration vectors $\ddot{\bar{\delta}}^k$ and $\ddot{\bar{\eta}}^k$ ($k \in \mathcal{F}$). This is accomplished by executing the "dynamic" call statement.

Dynamic Call Statement

CALL MRATE(NC, TH, TB, TS, FB, FS, TF, FF, GM, GMD, GMDD, DT, ET, WO, WDOT, DTD, ETD, HM)

where

NC = the integer n_c = number of system connections.

TH(i) = array containing the hinge torques, τ_i ; $i \in \mathcal{P}$.

TB(j) = array containing the elements of T^0 ; $j = 1, 2, 3$.

TS(i, j) = array containing the elements of T^i; $i \in \mathcal{P}^r, j = 1, 2, 3$.

FB(j) = array containing the elements of F^0 ; $j = 1, 2, 3$.

FS(i, j) = array containing the elements of F^i ; $i \in \mathcal{P}^r, j = 1, 2, 3$.

TF(n, i, j) = array containing the torque elements of λ^k ; $n = 1, \ldots, n_f$, $k = F(n, 1), i = 1, \ldots, n_k, j = 1, 2, 3$.

FF(n, i, j) = array containing the force elements of λ^k ; $n = 1, \ldots, n_f$, $k = F(n, 1), i = 1, \ldots, n_k, j = 1, 2, 3$.

GM(i) = array of angles, γ_i; $i \in \mathcal{P}$.

GMD(i) = array of the angular velocities, $\dot{\gamma}_i$; $i \in \mathcal{P}$.

GMDD(i) = array of the prescribed angular accelerations, $\ddot{\gamma}_i$; $i \in \mathcal{P}$.

DT(n, i) = array of appendage modal coordinates, $\bar{\delta}^k$; $n = 1, \ldots, n_f$, $k = F(n, 1), i = 1, \ldots, N_k$.

ET(n, i) = array of appendage modal coordinates, $\bar{\eta}^k$; $n = 1, \ldots, n_f$, $k = F(n, 1), i = 1, \ldots, N_k$.

WO(j) = array containing the components of ω^0 ; $j = 1, 2, 3$.

WDOT(j) = solution vector containing the elements of $\dot{\omega}^0, \ddot{\gamma}_1, \ldots, \ddot{\gamma}_n$; $j = 1, \ldots, n + 3$. (WDOT(1) = $\dot{\omega}_1^0$, WDOT (2) = $\dot{\omega}_2^0$, WDOT (3) = $\dot{\omega}_3^0$, WDOT (4) = $\ddot{\gamma}_1, \ldots$, WDOT ($n + 3$) = $\ddot{\gamma}_n$.)

DTD(n, i) = solution matrix for $\ddot{\bar{\delta}}^k$; $n = 1, \ldots, n_f, k = F(n, 1), i = 1, \ldots, N_k$.

ETD(n, i) = solution matrix for $\dot{\bar{\eta}}^k$; $n = 1, \ldots, n_f$, $k = F(n, 1)$, $i = 1, \ldots, N_k$.

HM = magnitude of the system angular momentum vector (see Appendix B for the momentum equations).

In summary, the call to MRATE supplies the subroutine with current instantaneous values for hinge torques and externally applied torques and forces on both rigid bodies and nonrigid appendages. Explicit expressions for computing these forcing functions, which may depend on γ_i, $\dot{\gamma}_i$, and other system or control variables, are located in the main calling program (see sample problem that follows). Current values of $\boldsymbol{\omega}^0$, γ_i, $\dot{\gamma}_i$, $\bar{\delta}^k$, and $\bar{\eta}^k$ are continuously produced by the main program's numerical integration operators and are therefore always available for input to MBDYFR.

It should be noted here that MBDYFR *does not* incorporate the terms in Eq. (42) that describe symmetric rotor torques on body $\mathscr{b}_k$. As a result, the *user* is required, if rotors are present, to supply these terms as part of a "new" T^k, i.e.,

$$T'^k = T^k - \tau_R^k - \tilde{\omega}^k \mathcal{J}^k(\tilde{\omega}^k + \dot{\psi}_R^k)$$

Thus, these terms must be formed in the main program along with Eq. (44), and T'^k is supplied to the subroutine as TB (if $k = 0$) or TS in the MRATE call statement.

Note also that, if any of the γ_i are to be prescribed, the appropriate values of γ_i, $\dot{\gamma}_i$, and $\ddot{\gamma}_i$ must be supplied to the subroutine by way of GM, GMD, GMDD, respectively, in the MRATE call statement. An example of this is shown in Section IVC.

When the MBDYFR subroutine is used, the main calling program must contain Fortran V (or IV) statements which specify "type" and allocate storage for the variables and arrays being used. The mandatory specification statements are listed here.

Required Specification Statements

INTEGER NC, NF, H$(n_c, 2)$, F$(n_f, 3)$, PI$(n + 1)$

REAL MB(7), MS$(n_c, 7)$, PB$(n_c, 3)$, PS$(n_c, n_c, 3)$, G$(n, 3)$, TH(n), TB(3), TS$(n_c, 3)$, FB(3), FS$(n_c, 3)$, GM(n), GMD(n), GMDD(n), ER$(n_f, 6n_k, N_k)$, EI$(n_f, 6n_k, N_k)$, MF$(n_f, n_k, 7)$, RF$(n_f, n_k, 3)$, WF(n_f, N_k), ZF(n_f, N_k), TF$(n_f, n_k, 3)$, FF$(n_f, n_k, 3)$, DT(n_f, N_k), ET(n_f, N_k), WO(3), SR$(n_f, 3)$

DOUBLE PRECISION WDOT$(n + 3)$, DTD(n_f, N_k), ETD(n_f, N_k)

Also, in order that storage allocation for arrays internal to MBDYFR be minimized, the following statement must appear *in the subroutine*:

PARAMETER QH = n, QC = n_c, QF = n_f, NK = n_k, NKT = N_k

The proper placement of this statement in MBDYFR is shown in the listing (Appendix C).

C. A Sample Problem Simulation

To illustrate the use of subroutine MBDYFR, the dynamical system shown in Fig. 3 will be simulated. It consists of a rigid central body, $\mathscr{b}_0$, to which is connected a rigid platform, $\mathscr{b}_2$, with 2 degrees of rotational freedom relative to $\mathscr{b}_0$. A spinning rotor, $\mathscr{b}_1$, is also connected to $\mathscr{b}_0$. The platform and the rotor each carry an elastic appendage, which will be modeled as a simple point mass supported by a massless elastic member.

For this test vehicle, the platform will be nominally nonrotating, while the rotor will have a nominal spin rate of ω_s about the spin axis fixed in $\mathscr{b}_0$. The appendage modal models must now be derived from the appropriate discrete coordinate equations.

Rotor Appendage Equations

The general appendage equation is Eq. (17), where the matrices M_k', G_k', and K_k' for the rotor substructure are as follows:

$$M^1 = \left[\begin{array}{ccc|c} m_1 & 0 & 0 & \\ 0 & m_1 & 0 & 0 \\ 0 & 0 & m_1 & \\ \hline & 0 & & 0 \end{array}\right] \qquad (6 \times 6)$$

$$\therefore \quad M_1' = M^1\left(U - \Sigma_{U0}\Sigma_{U0}^T \frac{M^1}{\mathfrak{M}}\right) = \left[\begin{array}{ccc|c} \mu_1 & 0 & 0 & \\ 0 & \mu_1 & 0 & 0 \\ 0 & 0 & \mu_1 & \\ \hline & 0 & & 0 \end{array}\right] \qquad (6 \times 6)$$

where

$$\mu_1 = m_1 - \frac{m_1^2}{\mathfrak{M}}$$

$$\mathfrak{M} = \mathfrak{M}_0 + \mathfrak{M}_1 + \mathfrak{M}_2$$

The rotor spin rate $= \Omega^s = [0 \quad 0 \quad \omega_s]^T$.

$$\therefore \quad G_1' = 2\left[\begin{array}{ccc|c} 0 & -\omega_s\mu_1 & 0 & \\ \omega_s\mu_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \\ \hline & 0 & & 0 \end{array}\right] \qquad (6 \times 6)$$

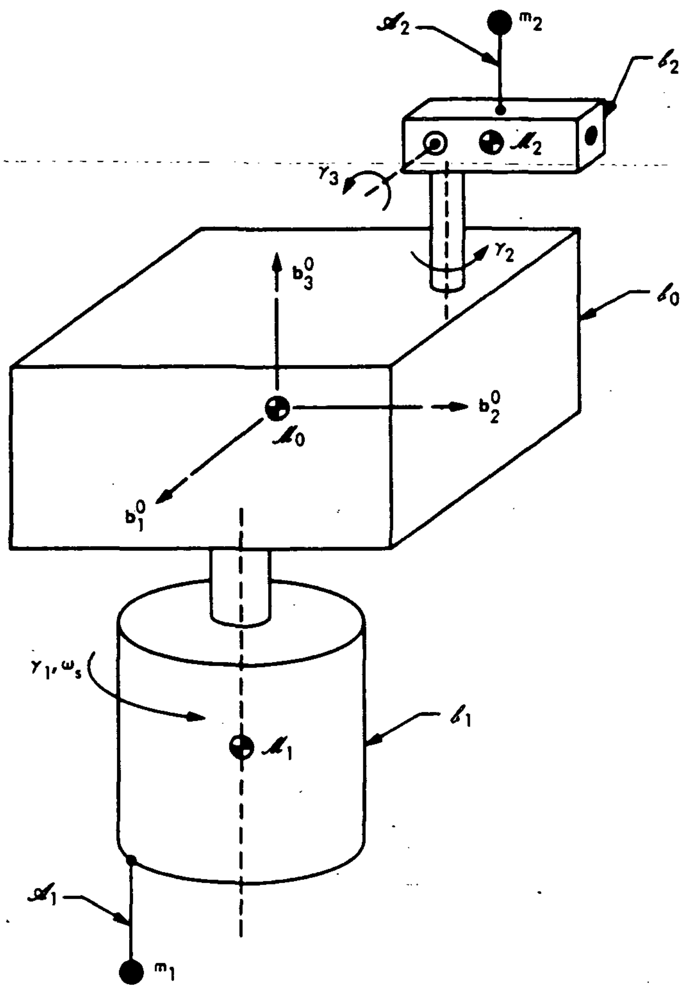

Fig. 3. MBDYFR simulation test vehicle

We will assume a symmetric stiffness matrix, K^1, of the form

$$K^1 = \left[\begin{array}{ccc|c} k_1 & 0 & 0 & \\ 0 & k_2 & 0 & 0 \\ 0 & 0 & k_3 & \\ \hline & 0 & & 0 \end{array}\right] \qquad (6 \times 6)$$

where k_1, k_2, and k_3 are the respective stiffness coefficients which restrain linear motion in the $\mathbf{b}_1^1$, $\mathbf{b}_2^1$, and $\mathbf{b}_3^1$ directions. Thus,

$$K_1' = \left[\begin{array}{ccc|c} k_1 - \omega_s^2\mu_1 & 0 & 0 & \\ 0 & k_2 - \omega_s^2\mu_1 & 0 & 0 \\ 0 & 0 & k_3 & \\ \hline & 0 & & 0 \end{array}\right]$$

The homogeneous rotor appendage equation may therefore be written as

$$\begin{bmatrix} \mu_1 & 0 & 0 \\ 0 & \mu_1 & 0 \\ 0 & 0 & \mu_1 \end{bmatrix} \ddot{q}^1 + 2 \begin{bmatrix} 0 & -\omega_s\mu_1 & 0 \\ \omega_s\mu_1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \dot{q}^1$$

$$+ \begin{bmatrix} k_1 - \mu_1\omega_s^2 & 0 & 0 \\ 0 & k_2 - \mu_1\omega_s^2 & 0 \\ 0 & 0 & k_3 \end{bmatrix} q^1 = 0$$

where $q^1 = [u_1^1 \; u_2^1 \; u_3^1]^T$ (realizing that $\beta_1^1 = \beta_2^1 = \beta_3^1 = 0$, since m_1 is a point mass).

If the equation is rewritten in first-order form, as in Eq. (18), it becomes

$$\mathcal{U}_1 \dot{Q}^1 + \mathcal{V}_1 Q^1 = 0$$

where

$$\mathcal{U}_1 = \left[\begin{array}{ccc|ccc} k_1 - \omega_s^2\mu_1 & 0 & 0 & & & \\ 0 & k_2 - \omega_s^2\mu_1 & 0 & & 0 & \\ 0 & 0 & k_3 & & & \\ \hline & & & \mu_1 & 0 & 0 \\ & 0 & & 0 & \mu_1 & 0 \\ & & & 0 & 0 & \mu_1 \end{array}\right]$$

$$\mathcal{V}_1 = \left[\begin{array}{ccc|ccc} & & & -k_1 + \omega_s^2\mu_1 & 0 & 0 \\ & 0 & & 0 & -k_2 + \omega_s^2\mu_1 & 0 \\ & & & 0 & 0 & -k_3 \\ \hline k_1 - \omega_s^2\mu_1 & 0 & 0 & 0 & -2\omega_s\mu_1 & 0 \\ 0 & k_2 - \omega_s^2\mu_1 & 0 & 2\omega_s\mu_1 & 0 & 0 \\ 0 & 0 & k_3 & 0 & 0 & 0 \end{array}\right]$$

and

$$Q^1 = [q^1 \mid \dot{q}^1]^T$$

The rotor appendage equation eigenvalues, λ_j, and corresponding eigenvectors, Φ_1^j, may then be found from

$$[\mathcal{U}_1\lambda_j + \mathcal{V}_1]\Phi_1^j = 0$$

From the characteristic equation, one finds that

$$\lambda_j = \pm i\left[\frac{k}{\mu_1} + \omega_s^2 \mp 2\omega_s \sqrt{\frac{k}{\mu_1}} \right]^{\frac{1}{2}}$$

and

$$\lambda_j = \pm i\left[\frac{k_3}{\mu_1} \right]^{\frac{1}{2}}$$

where $k = k_1 = k_2$.

If we now arbitrarily let $\sqrt{k/\mu_1} = 2\omega_s$ and $\sqrt{k_3/\mu_1} = 5\omega_s$, the eigenvalues become

$$\lambda_1 = i\omega_s$$

$$\lambda_2 = i3\omega_s$$

$$\lambda_3 = i5\omega_s$$

$$\lambda_4 = -i\omega_s$$

$$\lambda_5 = -i3\omega_s$$

$$\lambda_6 = -i5\omega_s$$

Note that the eigenvalues are imaginary as predicted and that they have been deliberately ordered to correspond to the form of Eq. (22), with conjugates in the lower half of Λ_1.

The eigenvectors corresponding to these eigenvalues may then be determined as

$$\Phi_1 = \left[\begin{array}{cccccc} i & -i & 0 & -i & i & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ \hline -\omega_s & 3\omega_s & 0 & -\omega_s & 3\omega_s & 0 \\ i\omega_s & i3\omega_s & 0 & -i\omega_s & -i3\omega_s & 0 \\ 0 & 0 & i5\omega_s & 0 & 0 & -i5\omega_s \end{array}\right] = \left[\begin{array}{c} \phi_1^j \\ \hline \phi_1^j\lambda_j \end{array}\right]$$

Also,

$$\Phi_1^{*T}\mathcal{U}_1\Phi_1 = \begin{bmatrix} 8\mu_1\omega_s^2 & & & & & \\ & 24\mu_1\omega_s^2 & & & 0 & \\ & & 50\mu_1\omega_s^2 & & & \\ & & & 8\mu_1\omega_s^2 & & \\ & 0 & & & 24\mu_1\omega_s^2 & \\ & & & & & 50\mu_1\omega_s^2 \end{bmatrix}$$

The final form of the appendage modal coordinate equations, shown in Eq. (26), can be obtained only if the eigenvectors are normalized so that $\Phi_1^{*T}\mathcal{U}_1\Phi_1 = U$, the diagonal unit matrix (see Ref. 3). Thus, succeeding columns in Φ_1 should be multiplied by $(8\mu_1\omega_s^2)^{-\frac{1}{2}}$, $(24\mu_1\omega_s^2)^{-\frac{1}{2}}$, $(50\mu_1\omega_s^2)^{-\frac{1}{2}}$, etc., for proper normalization in this case.

If we also arbitrarily truncate this modal transformation to just the first two modes, the resulting real and imaginary parts of $\bar{\phi}_1'$ become

$$\bar{\psi}_1 = \begin{bmatrix} 0 & 0 \\ \frac{1}{2\omega_s\sqrt{2\mu_1}} & \frac{1}{2\omega_s\sqrt{6\mu_1}} \end{bmatrix}, \quad \bar{\Gamma}_1 = \begin{bmatrix} \frac{1}{2\omega_s\sqrt{2\mu_1}} & -\frac{1}{2\omega_s\sqrt{6\mu_1}} \\ 0 & 0 \end{bmatrix}$$

Likewise,

$$\bar{\sigma}^1 = \begin{bmatrix} \omega_s & 0 \\ 0 & 3\omega_s \end{bmatrix}, \quad \bar{\xi}^1 = \begin{bmatrix} \xi_1^1 & 0 \\ 0 & \xi_2^1 \end{bmatrix}$$

Platform Appendage Equations

If the same process is applied to the nominally nonspinning platform appendage, its homogeneous equation of motion becomes

$$\begin{bmatrix} \mu_2 & 0 & 0 \\ 0 & \mu_2 & 0 \\ 0 & 0 & \mu_2 \end{bmatrix} \ddot{q}^2 + \begin{bmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & k_3 \end{bmatrix} q^2 = 0$$

Using the first-order equations again,

$$\mathcal{U}_2 \dot{Q}^2 + \mathcal{V}_2 Q^2 = 0$$

where

$$\mathcal{U}_2 = \left[\begin{array}{ccc|ccc} k_1 & 0 & 0 & & & \\ 0 & k_2 & 0 & & 0 & \\ 0 & 0 & k_3 & & & \\ \hline & & & \mu_2 & 0 & 0 \\ & 0 & & 0 & \mu_2 & 0 \\ & & & 0 & 0 & \mu_2 \end{array}\right]$$

$$\mathcal{V}_2 = \left[\begin{array}{ccc|ccc} & & & -k_1 & 0 & 0 \\ & 0 & & 0 & -k_2 & 0 \\ & & & 0 & 0 & -k_3 \\ \hline k_1 & 0 & 0 & & & \\ 0 & k_2 & 0 & & 0 & \\ 0 & 0 & k_3 & & & \end{array}\right]$$

and

$$Q^2 = [q^2 \mid \dot{q}^2]^T, \qquad \mu_2 = m_2 - \frac{m_2^2}{\mathfrak{M}}$$

one can easily determine that the eigenvalues are

$$\lambda_j = \pm i\sqrt{\frac{k_1}{\mu_2}}\ , \pm i\sqrt{\frac{k_2}{\mu_2}}\ , \pm i\sqrt{\frac{k_3}{\mu_2}}$$

If we let $k = k_1 = k_2 = k_3$, and $\sqrt{k/\mu_2} = \sigma_2$, then

$$\Lambda_2 = \begin{bmatrix} \sigma_2 i & & & & & \\ & \sigma_2 i & & & 0 & \\ & & \sigma_2 i & & & \\ & & & -\sigma_2 i & & \\ & 0 & & & -\sigma_2 i & \\ & & & & & -\sigma_2 i \end{bmatrix}$$

and

$$\Phi_2 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ \sigma_2 i & 0 & 0 & -\sigma_2 i & 0 & 0 \\ 0 & \sigma_2 i & 0 & 0 & -\sigma_2 i & 0 \\ 0 & 0 & \sigma_2 i & 0 & 0 & -\sigma_2 i \end{bmatrix} = \left[\begin{array}{c} \phi_2^j \\ \hline \phi_2^j \lambda_j \end{array}\right]$$

The appropriate normalization factor for each ϕ_2^i is $(2\mu_2\sigma_2^2)^{-\frac{1}{2}}$. Thus, if the platform appendage modal model is truncated to the first two (transverse bending) modes, the needed quantities are

$$\bar{\psi}_2 = \begin{bmatrix} \frac{1}{\sigma_2\sqrt{2\mu_2}} & 0 \\ 0 & \frac{1}{\sigma_2\sqrt{2\mu_2}} \end{bmatrix}, \quad \bar{\Gamma}_2 = 0$$

$$\bar{\sigma}^2 = \begin{bmatrix} \sigma_2 & 0 \\ 0 & \sigma_2 \end{bmatrix}, \quad \bar{\xi}^2 = \begin{bmatrix} \xi_1^2 & 0 \\ 0 & \xi_2^2 \end{bmatrix}$$

Test Vehicle Constants

To complete the specification of the test configuration shown in Fig. 3, numerical values can now be assigned to its various mass properties and other physical constants. First, let

$$\mathfrak{M}_0 = 399.9 \text{ kg}$$

$$\mathfrak{M}_1 = 50.1 \text{ kg}$$

$$\mathfrak{M}_2 = 50.0 \text{ kg}$$

$$m_1 = 1.0 \text{ kg}$$

$$m_2 = 5.0 \text{ kg}$$

$$\bar{J}^0 = \begin{bmatrix} 250. & 0. & 0. \\ 0. & 275. & 0. \\ 0. & 0. & 350. \end{bmatrix}, \text{kg-m}^2$$

$$\bar{J}^1 = \begin{bmatrix} 10. & 0. & 0. \\ 0. & 10. & 0. \\ 0. & 0. & 20. \end{bmatrix}, \text{kg-m}^2$$

$$\therefore \quad \mathfrak{M} = \mathfrak{M}_0 + \mathfrak{M}_1 + \mathfrak{M}_2 = 500.0 \text{ kg}$$

$$\therefore \quad \mu_1 = .998 \text{ kg}$$

$$\therefore \quad \mu_2 = 4.95 \text{ kg}$$

$$\bar{J}^2 = \begin{bmatrix} 6. & 0 & 0 \\ 0 & 3. & 0 \\ 0 & 0 & 8. \end{bmatrix}, \text{kg-m}^2$$

Also, let

$$\omega_s = 10. \text{ rad/s}, \quad \xi_1^1 = \xi_2^1 = .01$$

$$\sigma_2 = 9. \text{ rad/s}, \quad \xi_1^2 = \xi_2^2 = .01$$

$$\therefore \quad \bar{\psi}_1 = \begin{bmatrix} 0 & 0 \\ .035391 & .020433 \end{bmatrix}, \quad \bar{\Gamma}_1 = \begin{bmatrix} .035391 & -.020433 \\ 0 & 0 \end{bmatrix}$$

$$\bar{\psi}_2 = \begin{bmatrix} .035313 & 0 \\ 0 & .035313 \end{bmatrix}, \quad \bar{\Gamma}_2 = 0$$

The locations of the two point masses (see Figs. 4 and 5) relative to their substructure's mass center when they are in the nominal deformed state will be assumed as

$$r_1 = [.33 \quad 0 \quad -.493]^T \text{ meters}$$

$$r_2 = [0 \quad 0 \quad .56]^T \text{ meters}$$

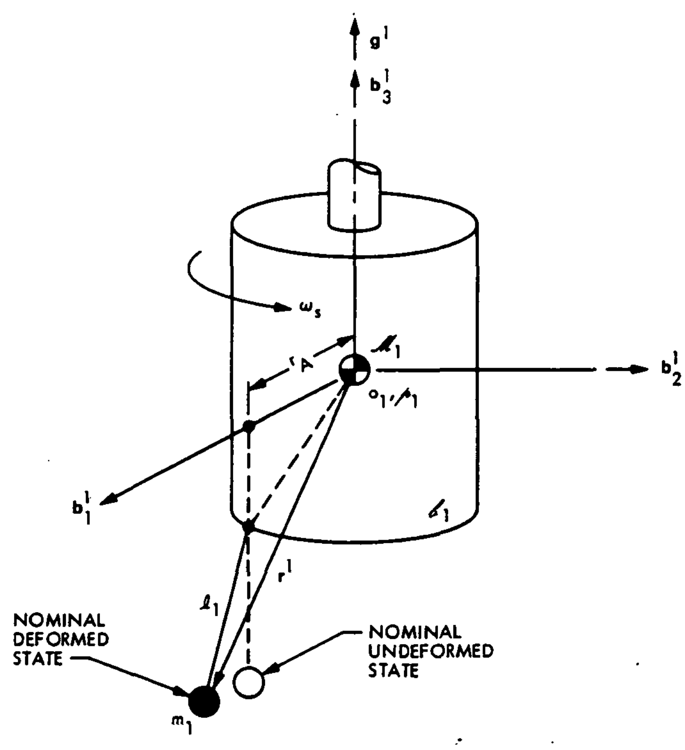

Fig. 4. Substructure $\mathscr{s}_1$

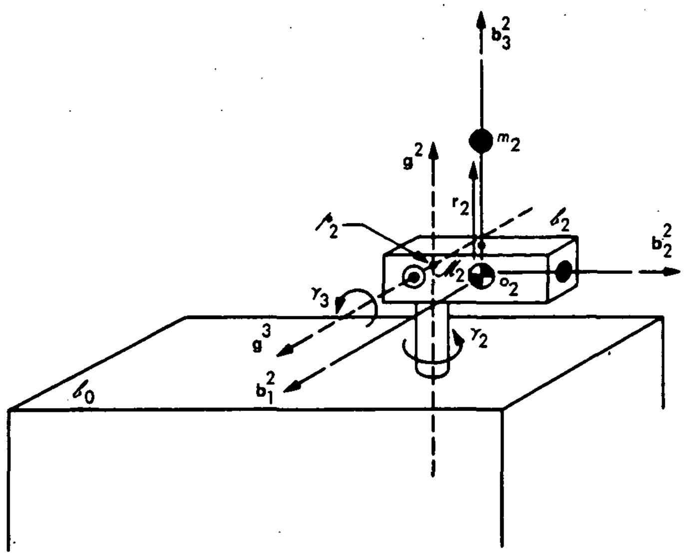

Fig. 5. Substructure s_2

Locations for the interbody connections, relative to substructure mass centers, are

$$p^{01} = [0. \quad 0. \quad -2.]^T \text{ meters}$$

$$p^{02} = [0. \quad 1. \quad 1.]^T \text{ meters}$$

$$p^{10} = [0. \quad 0. \quad 0.] \text{ meters}$$

$$p^{20} = [0. \quad -.3 \quad 0.] \text{ meters}$$

The three hinge directions are given by the direction cosines

$$\text{rotor}: \quad g^1 = [0. \quad 0. \quad 1.]^T$$

$$\text{platform}: \quad g^2 = [0. \quad 0. \quad 1.]^T$$

$$\text{platform}: \quad g^3 = [1. \quad 0. \quad 0.]^T$$

Also,

$$n_c = 2, \quad n_f = 2, \quad n_1 = 1, \quad n_2 = 1, \quad N_1 = 2, \quad N_2 = 2$$

$$h_1 = 0, \quad h_2 = 0, \quad d_1 = 1, \quad d_2 = 2, \quad n = 3$$

As a result of these choices, the initializing call statement arguments become

$$\text{NC} = 2$$

$$\text{H} = \begin{bmatrix} 0 & 1 \\ 0 & 2 \end{bmatrix}$$

$$MB = [250. \quad 275. \quad 350. \quad 0. \quad 0. \quad 0. \quad 399.9]$$

$$MS = \begin{bmatrix} 10. & 10. & 20. & 0. & 0. & 0. & 50.1 \\ 6. & 3. & 8. & 0. & 0. & 0. & 50.0 \end{bmatrix}$$

$$PB = \begin{bmatrix} 0. & 0. & -2. \\ 0. & 1. & 1. \end{bmatrix}$$

$$PS(2, 2, j) = [0. \quad -.3 \quad 0.]$$ (all other PS elements are zero)

$$G = \begin{bmatrix} 0. & 0. & 1. \\ 0. & 0. & 1. \\ 1. & 0. & 0. \end{bmatrix}$$

$$PI = [0 \quad 0 \quad 0 \quad 1]$$ (assuming no prescribed hinge motions)

$$NF = 2$$

$$F = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 1 & 2 \end{bmatrix}$$

$$ER(1, i, j) = \begin{bmatrix} 0. & 0. \\ .035391 & .020433 \\ 0. & 0. \\ 0. & 0. \\ 0. & 0. \\ 0. & 0. \end{bmatrix}$$

$$EI(1, i, j) = \begin{bmatrix} .035391 & -.020433 \\ 0. & 0. \\ 0. & 0. \\ 0. & 0. \\ 0. & 0. \\ 0. & 0. \end{bmatrix}$$

$$\mathrm{ER}(2, i, j) = \begin{bmatrix} .035313 & 0. \\ 0. & .035313 \\ 0. & 0. \\ 0. & 0. \\ 0. & 0. \\ 0. & 0. \end{bmatrix}$$

$$\mathrm{EI}(2, i, j) = 0.$$

$$\mathrm{SR} = \begin{bmatrix} 0. & 0. & 10. \\ 0. & 0. & 0. \end{bmatrix}$$

$$\mathrm{MF}(1, 1, j) = [0. \quad 0. \quad 0. \quad 0. \quad 0. \quad 0. \quad 1.0]$$

$$\mathrm{MF}(2, 1, j) = [0. \quad 0. \quad 0. \quad 0. \quad 0. \quad 0. \quad 5.0]$$

$$\mathrm{RF}(1, 1, j) = [.3333 \quad 0. \quad -.4930]$$

$$\mathrm{RF}(2, 1, j) = [0. \quad 0. \quad .56]$$

$$\mathrm{WF} = \begin{bmatrix} 10. & 30. \\ 9. & 9. \end{bmatrix}$$

$$\mathrm{ZF} = \begin{bmatrix} .01 & .01 \\ .01 & .01 \end{bmatrix}$$

Test Vehicle Dynamics

Before simulating a specific dynamic case for the test vehicle of Fig. 3, the characteristics of the interbody connections must be defined. The connection between $\mathscr{b}_0$ and rotor $\mathscr{b}_1$ will be assumed a frictionless bearing so that

$$\tau_1 = 0$$

The platform hinge connections will be assumed to be of the linear spring-damper type, i.e.,

$$\tau_2 = -K_2(\gamma_2 - \gamma_{2c}) - B_2\dot{\gamma}_2$$

$$\tau_3 = -K_3(\gamma_3 - \gamma_{3c}) - B_3\dot{\gamma}_3,$$

where γ_{2c} and γ_{3c} are platform angular position commands. The values of the constants K_2, K_3, B_2, B_3 are arbitrarily chosen as

$$K_2 = 250.\ \text{n-m/rad}, \qquad B_2 = 50.\ \text{n-m-s/rad}$$

$$K_3 = 300.\ \text{n-m/rad}, \qquad B_3 = 50.\ \text{n-m-s/rad}$$

The dynamic response to be simulated here will be that due to a high-rate platform slew sequence. Slew commands γ_{2c} and γ_{3c} will be generated by integrating the time functions shown in Fig. 6. This will result in a 10-deg rotation about $\mathbf{g}^2$ and a 10-deg rotation about $\mathbf{g}^3$.

Initially, the rotor is spinning at 10 rad/s relative to $\mathscr{b}_0$, and the rotor appendage is at rest relative to the rotor but deflected radially outward in its steady-state deformed position. (One can show from Eq. (17), with the assumption $k/\mu_1 = 4\omega_s^2$, that the radial deformation (in the $\mathbf{b}_1^1$ direction) due to spin is $r_A/3$, where r_A is the distance from the rotor spin axis to the appendage attachment point.) The platform, $\mathscr{b}_1$, as well as the base body, $\mathscr{b}_0$, are initially at rest. At $t = 1$ s, the command is issued to rotate the platform about $\mathbf{g}^2$ at a rate of 10 deg/s until $t = 2$ s; again at $t = 3$ s, a command to rotate about the $\mathbf{g}^3$ axis at 10 deg/s appears and ends at $t = 4$ s. The computer simulation program, employing MBDYFR, for this dynamic maneuver is shown in Fig. 7.

Notice that the necessary dimension specifications for each variable are stated in the JPL CSSL III simulation language as: ARRAY MB(7), MS(2, 7), . . . , etc.

An auxiliary routine, called HCK, is used in the simulation to keep track of the rotations of the reference body ($\mathscr{b}_0$) relative to an inertially fixed frame. HCK uses Euler parameters to do this, and it is initialized using Euler angles. The variable, THET, is calculated in the program by means of HCK and represents the angular deviation of the $\mathbf{b}_3^0$ axis from its initial, inertially fixed position, i.e., the reference body "nutation" angle.

The CSSL III function, "STEP," provides the unit step function when the independent variable, TIME, is greater than the specified constant. "INTEG(a_1, a_2) signifies the integration of a_1 with respect to TIME, where a_2 is the initial condition.

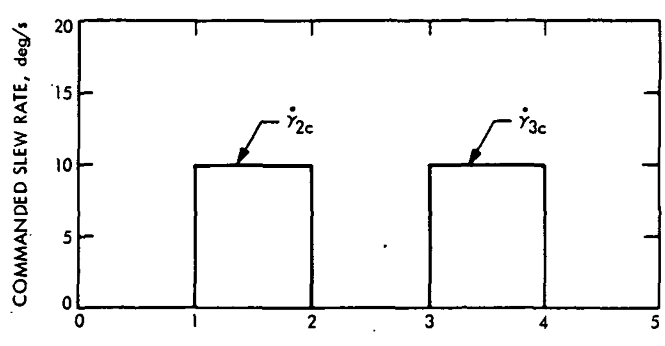

Fig. 6. Commanded slew rates

```
CSSL III  JET PROPULSION LABORATORY  040374-A002H  021775-021636

*** START                    T(RUN)=  28.032 T(TASK)=     .003  CTP    =     .760
                                             DT(TASK)=     .003 DCTP    =     .760

    PROGRAM   3-BODY VEHICLE WITH SPINNING ROTOR AND 2 FLEXIBLE APPENDAGES
    *SC4020   BLDG/198,BOX/601,CAMERA/9IN,FRAMES/50
          ARRAY MB(7),MS(2,7),PB(2,3),PS(2,2,3),G(3,3),TH(3),TB(3),TS(2,3)
          ARRAY FB(3),FS(2,3),GM(3),GMD(3),GMDD(3),ER(2,6,2),EI(2,6,2)
          ARRAY MF(2,1,7),RF(2,1,3),WF(2,2),ZF(2,2),TF(2,1,3),FF(2,1,3)
          ARRAY SR(2,3),DT(2,2),ET(2,2),WO(3),U(2,1,3),UD(2,1,3)
          DOUBLE PRECISION WDOT(6),DTD(2,2),ETD(2,2),EC(14)
          INTEGER NC,NF,H(2,2),F(2,3),PI(4)
          DATA H/0,0,1,2/PI/0,0,0,1/
          DATA MB/250.,275.,350.,0.,0.,0.,399.9/
          DATA MS(1,1)/10./MS(1,2)/10./MS(1,3)/20./MS(1,7)/50.1/
          DATA MS(2,1)/6./MS(2,2)/3./MS(2,3)/8./MS(2,7)/50./
          DATA PB(1,1)/0./PB(1,2)/0./PB(1,3)/-2./
          DATA PB(2,1)/0./PB(2,2)/1./PB(2,3)/1./
          DATA PS(2,2,1)/0./PS(2,2,2)/-.3/PS(2,2,3)/0./
          DATA G(1,3)/1./G(2,3)/1./G(3,1)/1./
          DATA F/1,2,1,1,2,2/
          DATA ER(1,2,1)/.03539075/ER(1,2,2)/.02043286/
          DATA EI(1,1,1)/.03539075/EI(1,1,2)/-.02043286/
          DATA ER(2,1,1)/.03531343/ER(2,2,2)/.03531343/
          DATA RF(1,1,1)/.3333/RF(1,1,3)/-.4930/RF(2,1,3)/.56/
          DATA MF(1,1,7)/1.0/MF(2,1,7)/5.0/
          DATA ZF(1,1)/.01/ZF(1,2)/.01/ZF(2,1)/.01/ZF(2,2)/.01/
          DATA WF(1,1)/10./WF(1,2)/30./WF(2,1)/9.0/WF(2,2)/9.0/
          DATA SR(1,1)/0./SR(1,2)/0./SR(1,3)/10./
          CONSTANT K2=250.,B2=50.,K3=300.,B3=50.
          CONSTANT TFINAL=10.,CLKTIM=900.,PIE=3.14159265
          CONSTANT PHIZ=0.,THETZ=0.,PSIZ=0.
          CONSTANT GM1DI=10.,W1I=0.,W2I=0.,W3I=0.
    INITIAL
          NC=2   $   NF=2
          CALL MBDYFR(NC,H,MB,MS,PB,PS,G,PI,NF,F,ER,EI,SR,MF,RF,WF,ZF)
          PZI,PAI,PBI,PCI=HCK(INITL,PHIZ,THETZ,PSIZ)
    END
    DYNAMIC
          IF(TIME.GE.TFINAL) GO TO S1
    DERIVATIVE   BODY3F
          VARIABLE TIME=0.   $   CINTERVAL  CI=.01
          XERROR W1=1.E-9,W2=1.E-9,W3=1.E-9,GM2D=1.E-9,GM3D=1.E-9,...
          GM2=1.E-9,GM3=1.E-9,PA0=1.E-9,PB0=1.E-9,PC0=1.E-9,GM1=1.E-5
          MERROR W1=1.E-9,W2=1.E-9,W3=1.E-9,GM2D=1.E-9,GM3D=1.E-9,...
          GM2=1.E-9,GM3=1.E-9,PA0=1.E-9,PB0=1.E-9,PC0=1.E-9,GM1=1.E-5
          STPCLK   CLKTIM
          OUTPUT 10,W1D,W2D,W3D,W1,W2,W3,GM1D,GM2D,GM3D,GM1,GM2,GM3,THET,...
                 DT1A,DT1B,DT2A,DT2B,ET1A,ET1B,ET2A,ET2B,PZ0,PA0,PB0,...
                 PC0,U1X,U1Y,U1Z,U2X,U2Y,U2Z,U1XD,U1YD,U2XD,U2YD,GM2C,...
                 GM3C,GM2CD,GM3CD,ANGM
          PREPAR THET,W1,W2,W3,GM1D,GM2D,GM3D,GM2,GM3,U1X,U1Y,U2X,U2Y,...
                 GM1,U1XD,U1YD,U2XD,U2YD,GM2C,GM3C,ANGM
    NOSORT
          WO(1)=W1 $ WO(2)=W2 $ WO(3)=W3
          GM(1)=GM1 $ GM(2)=GM2 $ GM(3)=GM3
```

Fig. 7. Simulation program for test vehicle dynamics using MBDYFR

ORIGINAL PAGE IS
OF POOR QUALITY

```
      GMD(1)=GM1D $ GMD(2)=GM2D $ GMD(3)=GM3D
      DT(1,1)=DT1A $ DT(1,2)=DT1B $ ET(1,1)=ET1A $ ET(1,2)=ET1B
      DT(2,1)=DT2A $ DT(2,2)=DT2B $ ET(2,1)=ET2A $ ET(2,2)=ET2B
COMMENT...
                 PLATFORM POSITION COMMANDS
COMMENT
      GM2CD=(STEP(1.0,TIME)-STEP(2.0,TIME))*PIE*10./180.
      GM3CD=(STEP(3.0,TIME)-STEP(4.0,TIME))*PIE*10./180.
      GM2C=INTEG(GM2CD,0.)    $    GM3C=INTEG(GM3CD,0.)
COMMENT...
                 REFERENCE BODY NUTATION ANGLE
COMMENT
      D1,D2=HCK(MATRIX,PZO,PAO,PBO,PCO)
      DC1,DC2,DC3=HCK(BTOI,0.,0.,1.,D1,D2)
      DCM=SQRT(DC1**2 + DC2**2)
      THET=ASIN(DCM)*180./PIE
COMMENT...
                 HINGE TORQUES
COMMENT
      TH(2)=-K2*(GM2-GM2C) - B2*GM2D
      TH(3)=-K3*(GM3-GM3C) - B3*GM3D
COMMENT...
                 SYSTEM ANGULAR ACCELERATIONS
COMMENT
      CALL MRATE(NC,TH,TB,TS,FB,FS,TF,FF,GM,GMD,GMDD,DT,ET,WO,WDOT,...
            DTD,ETD,HM,U,UD)
      U1XD=UD(1,1,1) $ U1YD=UD(1,1,2) $ U2XD=UD(2,1,1) $ U2YD=UD(2,1,2)
      U1X=U(1,1,1) $ U1Y=U(1,1,2) $ U1Z=U(1,1,3)
      U2X=U(2,1,1) $ U2Y=U(2,1,2) $ U2Z=U(2,1,3)
      W1D=WDOT(1)  $   W2D=WDOT(2)   $   W3D=WDOT(3)   $   ANGM=HM
COMMENT...
                 SYSTEM ANGULAR RATES AND POSITIONS
COMMENT
      W1=INTEG(WDOT(1),W1I)
      W2=INTEG(WDOT(2),W2I)
      W3=INTEG(WDOT(3),W3I)
      GM1D=INTEG(WDOT(4),GM1DI)
      GM2D=INTEG(WDOT(5),0.)
      GM3D=INTEG(WDOT(6),0.)
      GM1=INTEG(GM1D,0.) $ GM2=INTEG(GM2D,0.) $ GM3=INTEG(GM3D,0.)
      DT1A=INTEG(DTD(1,1),0.)  $  DT1B=INTEG(DTD(1,2),0.)
      DT2A=INTEG(DTD(2,1),0.)  $  DT2B=INTEG(DTD(2,2),0.)
      ET1A=INTEG(ETD(1,1),0.)  $  ET1B=INTEG(ETD(1,2),0.)
      ET2A=INTEG(ETD(2,1),0.)  $  ET2B=INTEG(ETD(2,2),0.)
COMMENT...
                 HCK PARAMETER RATES AND POSITIONS
COMMENT
      PZOD,PAOD,PBOD,PCOD=HCK(HCK,PZO,PAO,PBO,PCO,W1,W2,W3)
      PZO=INTEG(PZOD,PZI)   $   PAO=INTEG(PAOD,PAI)
      PBO=INTEG(PBOD,PBI)   $   PCO=INTEG(PCOD,PCI)
END
END
END
TERMINAL
S1..    CONTINUE
END
END
```

Fig. 7 (contd)

ORIGINAL PAGE IS
OF POOR QUALITY

All arithmetic statements are in Fortran, although CSSL III allows several statements in a single line if separated by a "\$". Variables to be plotted at every communication interval, CI, are listed in the PREPAR statement. Printed variables are listed in the OUTPUT statement.

The statement "CALL MBDYFR(NC, H, . . .)" is located in the INITIAL section and is therefore executed only once, i.e., prior to the dynamic calculations. However, "CALL MRATE(NC, . . .)" is in the DERIVATIVE section and is thus executed at every integration step. Note that two *additional* output variables have been added to the MRATE call statement argument list. They are U and UD, containing the appendage deformations u_1^1, u_2^1, u_3^1, u_1^2, . . . etc. and the deformation rates $\dot{u}_1^1$, $\dot{u}_2^1$, . . . , respectively. These variables are always available internal to MBDYFR using the relations of Eq. (27) and are outputted here only to more clearly illustrate the dynamic response of the system. (β_1^1, β_2^1, . . . , $\dot{\beta}_1^1$, $\dot{\beta}_2^1$, . . . etc. could also be obtained from the subroutine in those cases where the appendage nodal bodies have inertia.)

Results of the dynamic simulation are shown in the computer plots of Fig. 8, and the sample printout is presented in Fig. 9.

The solutions show, as expected, that all three components of the reference body angular velocity, $\boldsymbol{\omega}^0$, are strongly perturbed by the platform as it accelerates or decelerates. Further, induced vibrations of the platform appendage are also in evidence on the reference body rates. Rotor spin rate, $\dot{\gamma}_1$, relative to $\boldsymbol{b}_0$ remains very close to its initial and nominal value of 10 rad/s, although the effect of slewing the platform about an axis parallel to rotor spin is quite evident as are the subsequent vibrations due to platform appendage motion. Platform hinge rates, $\dot{\gamma}_2$ and $\dot{\gamma}_3$, also show some appendage vibration, although it is very small compared to the slewing rate transients.

The components of rotor appendage deformation u_1^1, u_2^1 exhibit both modal frequencies, ω_s and $3\omega_s$, but are relatively small in amplitude compared to the platform appendage deflections u_1^2, u_2^2. An "$X-Y$" plot of the platform appendage's deflections relative to its locally fixed coordinate frame is also shown.

System angular momentum magnitude in this test simulation should remain constant since no external forces or torques are being applied. The plot of HM shows this to be true very closely. Small deviations from a perfectly constant angular momentum in the simulations are to be expected due to the presence of modal damping (see Appendix A), numerical integration error, and round-off error.

N76 12094

IV. Systems With Nonrotating Appendages

A. Equations

In Part III, dynamical equations were developed for the substructure tree on the basis of (1) arbitrarily small flexible appendage deformations (and rates) from some nominal state and (2) arbitrarily small deviations of the angular rate of any rigid appendage base from a constant nonzero spin rate, Ω^k. In this section, the assumption will be made that $\Omega^k = 0$ ($k \in \mathscr{F}$), i.e., that the appendage bases are nonrotating.

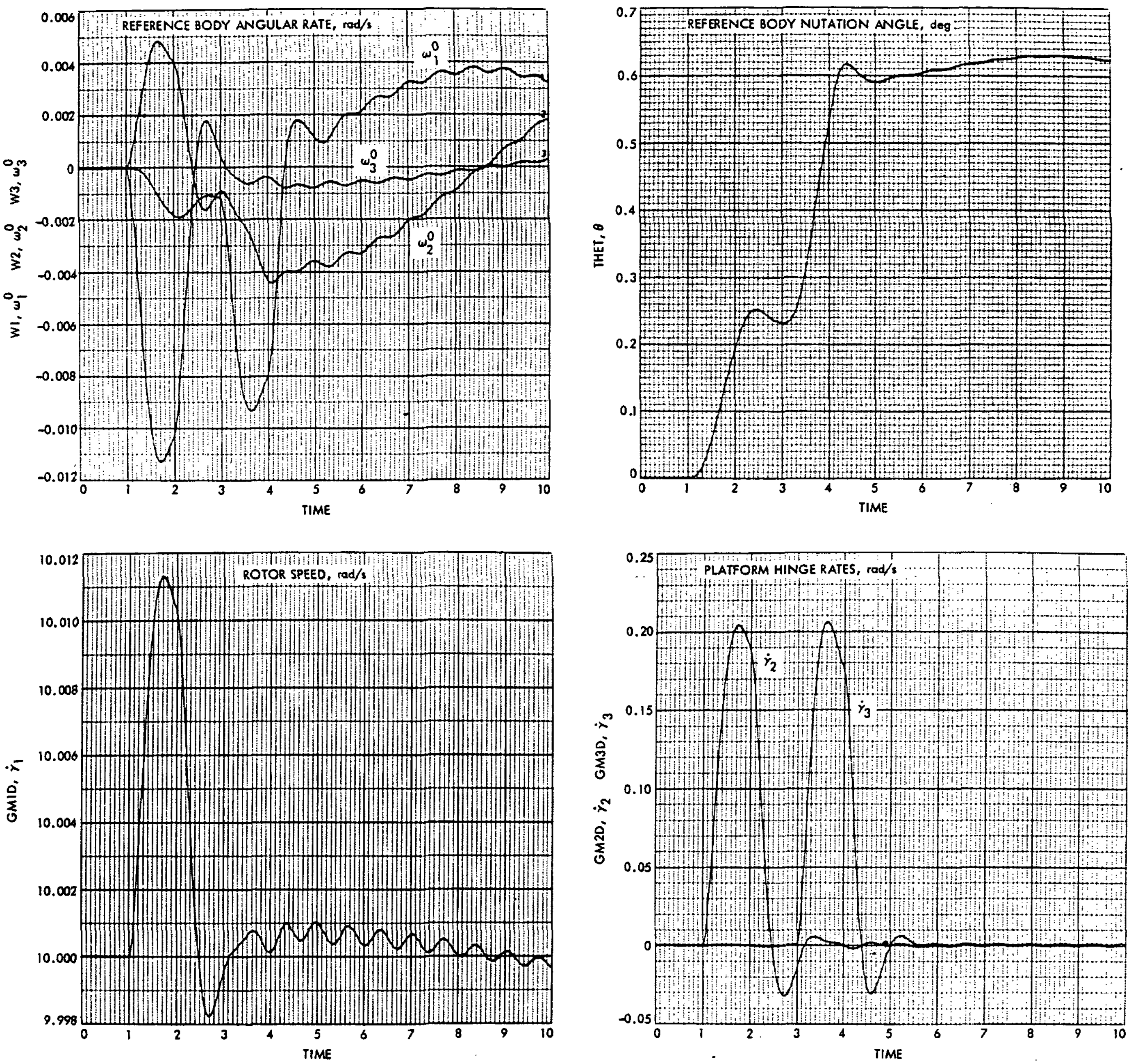

Fig. 8. Test vehicle (with spinning rotor) simulation results using MBDYFR

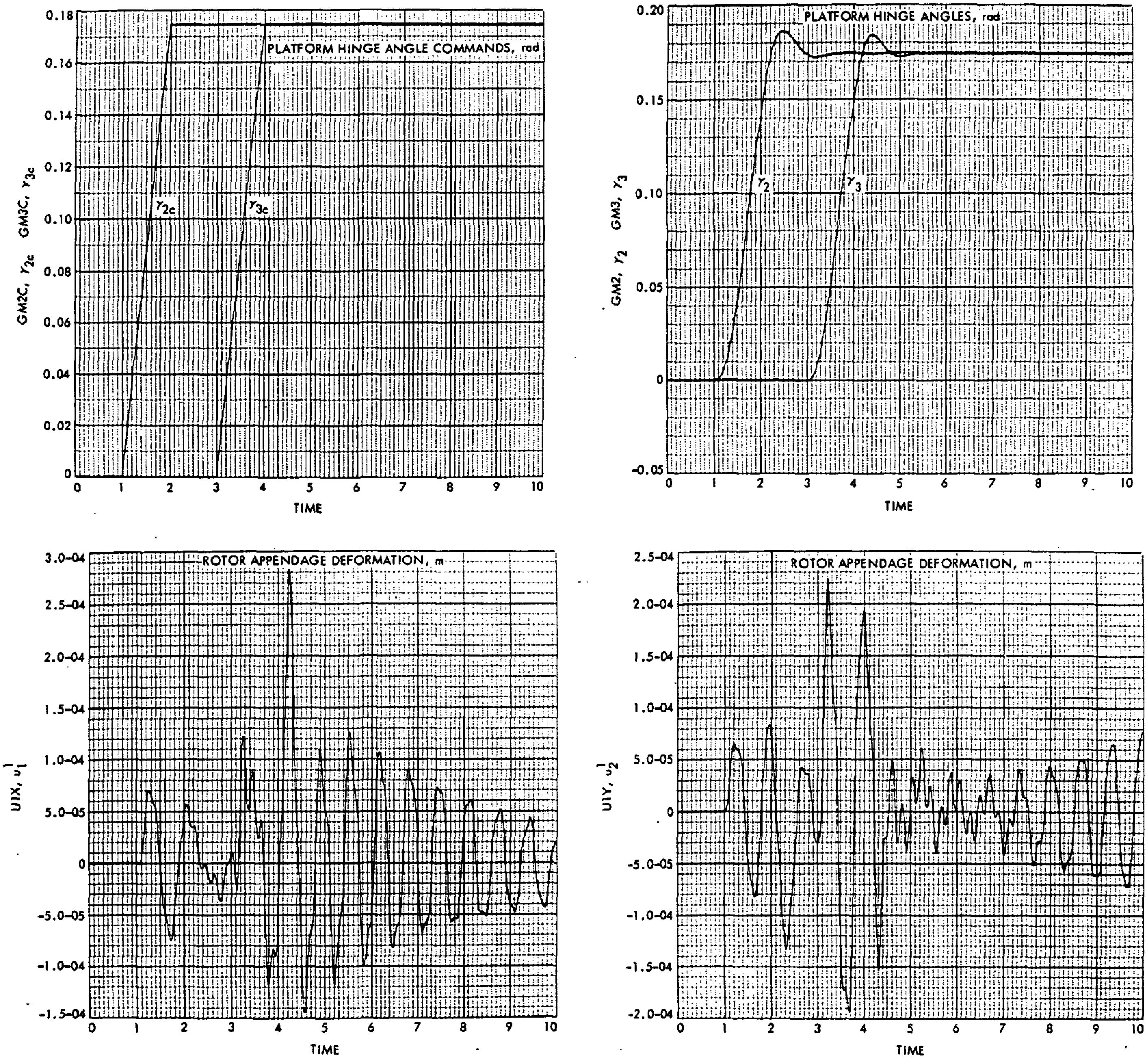
PLATFORM HINGE ANGLE COMMANDS, rad
GM2C, γ_{2c} GM3C, γ_{3c}
γ_{2c}
γ_{3c}
TIME
PLATFORM HINGE ANGLES, rad
GM2, γ_2 GM3, γ_3
γ_2
γ_3
TIME
ROTOR APPENDAGE DEFORMATION, m
U1X, v_1^1
TIME
ROTOR APPENDAGE DEFORMATION, m
U1Y, v_2^1
TIME

Fig. 8 (contd)

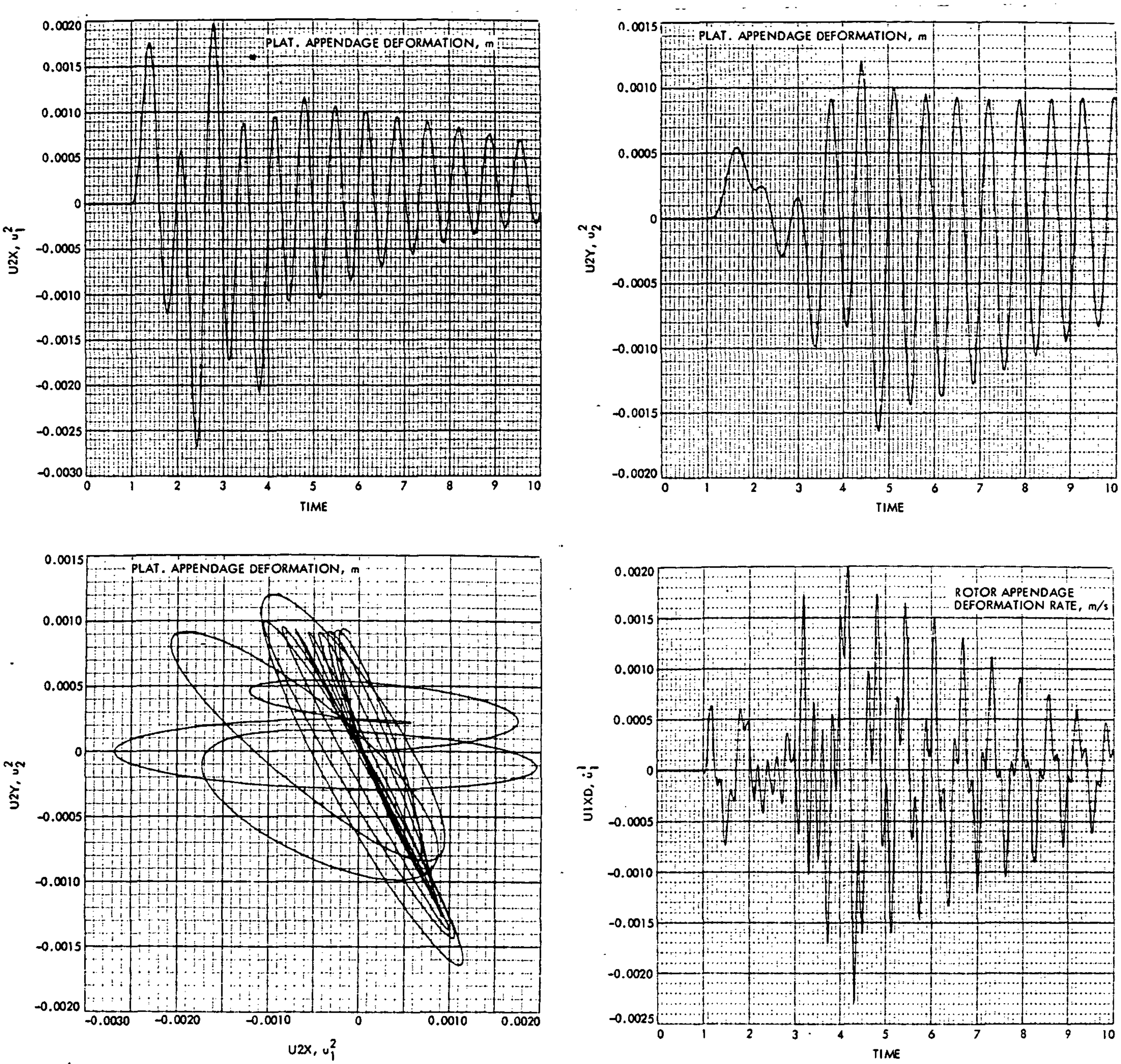
PLAT. APPENDAGE DEFORMATION, m
U2X, u_1^2
TIME
PLAT. APPENDAGE DEFORMATION, m
U2Y, u_2^2
TIME
PLAT. APPENDAGE DEFORMATION, m
U2Y, u_2^2
U2X, u_1^2
ROTOR APPENDAGE DEFORMATION RATE, m/s
U1XD, $\dot{u}_1^1$
TIME

Fig. 8 (contd)

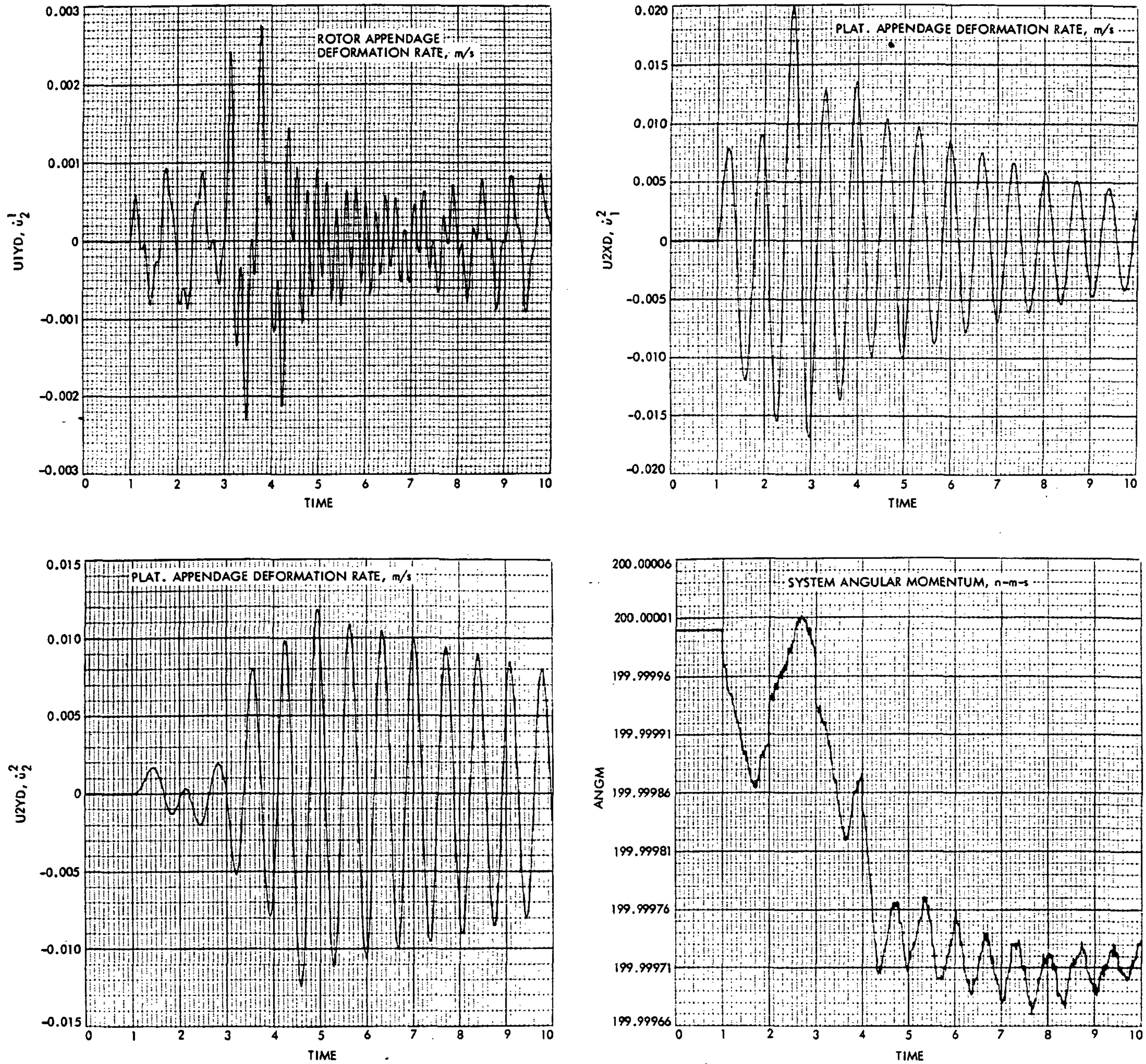
ROTOR APPENDAGE DEFORMATION RATE, m/s
U1YD, $\dot{u}_2^1$
0.003
0.002
0.001
0
-0.001
-0.002
-0.003
TIME
PLAT. APPENDAGE DEFORMATION RATE, m/s
U2XD, $\dot{u}_1^2$
0.020
0.015
0.010
0.005
0
-0.005
-0.010
-0.015
-0.020
TIME
PLAT. APPENDAGE DEFORMATION RATE, m/s
U2YD, $\dot{u}_2^2$
0.015
0.010
0.005
0
-0.005
-0.010
-0.015
TIME
SYSTEM ANGULAR MOMENTUM, n-m-s
ANGM
200.00006
200.00001
199.99996
199.99991
199.99986
199.99981
199.99976
199.99971
199.99966
TIME
0 1 2 3 4 5 6 7 8 9 10

Fig. 8 (contd)

TIME	= 9.30000	W1D	= -1.236511-03	W2D	= 6.946611-04	W3D	= 6.680829-04
		W1	= 3.556527-03	W2	= 9.172441-04	W3	= 1.222779-04
		GM1D	= 9.99981	GM2D	= 3.833038-04	GM3D	= 3.453655-04
		GM1	= 93.0119	GM2	= .174599	GM3	= .174527
		THET	= .628172	DT1A	= 5.599171-04	DT1B	= 4.708252-04
		DT2A	= -2.450952-03	DT2B	= 1.272927-02	ET1A	= -2.991362-04
		ET1B	= 3.456706-05	ET2A	= -4.290413-03	ET2B	= 2.997854-03
		PZO	= .999969	PAO	= 1.258746-03	PBO	= -5.335321-03
		PCO	= -6.043045-03	U1X	= 2.258592-05	U1Y	= 5.887238-05
		U1Z	= 0.000000	U2X	= -1.731031-04	U2Y	= 8.990284-04
		U2Z	= 0.000000	U1XD	= 1.809007-04	U1YD	= 1.693549-04
		U2XD	= 2.727165-03	U2YD	= -1.905561-03	GM2C	= .174533
		GM3C	= .174533	GM2CD	= 0.000000	GM3CD	= 0.000000
		ANGM	= 200.000				
TIME	= 9.40000	W1D	= -5.872507-04	W2D	= 1.192746-03	W3D	= 4.019024-04
		W1	= 3.461807-03	W2	= 1.007584-03	W3	= 1.778381-04
		GM1D	= 9.99971	GM2D	= 2.937455-04	GM3D	= 3.525509-04
		GM1	= 94.0119	GM2	= .174635	GM3	= .174564
		THET	= .627909	DT1A	= 5.533947-04	DT1B	= 4.892285-04
		DT2A	= 3.079498-03	DT2B	= 5.541556-03	ET1A	= -2.872588-04
		ET1B	= 4.543387-04	ET2A	= -7.094131-03	ET2B	= 1.162703-02
		PZO	= .999969	PAO	= 1.434171-03	PBO	= -5.288482-03
		PCO	= -6.034436-03	U1X	= 3.889949-05	U1Y	= 5.916278-05
		U1Z	= 0.000000	U2X	= 2.174952-04	U2Y	= 3.913827-04
		U2Z	= 0.000000	U1XD	= 2.080791-04	U1YD	= -3.536803-04
		U2XD	= 4.509326-03	U2YD	= -7.390628-03	GM2C	= .174533
		GM3C	= .174533	GM2CD	= 0.000000	GM3CD	= 0.000000
		ANGM	= 200.000				
TIME	= 9.50000	W1D	= 8.668784-05	W2D	= 1.815489-03	W3D	= 4.299192-07
		W1	= 3.439150-03	W2	= 1.160164-03	W3	= 1.977724-04
		GM1D	= 9.99973	GM2D	= -2.058826-05	GM3D	= 1.211780-04
		GM1	= 95.0119	GM2	= .174650	GM3	= .174589
		THET	= .627575	DT1A	= 2.189948-06	DT1B	= -4.258137-04
		DT2A	= 8.515822-03	DT2B	= -5.457229-03	ET1A	= 1.589932-05
		ET1B	= 6.865656-04	ET2A	= -4.239752-03	ET2B	= 1.108587-02
		PZO	= .999969	PAO	= 1.606681-03	PBO	= -5.235608-03
		PCO	= -6.023887-03	U1X	= 2.693162-05	U1Y	= -1.724618-05
		U1Z	= 0.000000	U2X	= 6.014458-04	U2Y	= -3.854269-04
		U2Z	= 0.000000	U1XD	= -5.235856-04	U1YD	= -8.529636-04
		U2XD	= 2.694963-03	U2YD	= -7.046642-03	GM2C	= .174533
		GM3C	= .174533	GM2CD	= 0.000000	GM3CD	= 0.000000
		ANGM	= 200.000				

Fig. 9. Simulation printout for test vehicle with spinning rotor

ORIGINAL PAGE IS OF POOR QUALITY

Equation (29) may now be simplified by the assumptions (for $k \in \mathcal{F}$) $\omega^k \approx 0$, $\dot{\omega}^k \approx 0$, $q^k \approx 0$, $\dot{q}^k \approx 0$, $\ddot{q}^k \approx 0$, $c^k \approx 0$, $\dot{c}^k \approx 0$, to obtain

$$
\begin{aligned}
(k \in \mathcal{B}) \quad W^k = {} & T^k + \sum_{r \in \mathcal{B}} \tilde{D}^{kr} C^{kr} F^r + \left[\tilde{F}^k - \left(C^{k0} \frac{\mathfrak{M}_k}{\mathfrak{M}} F \right)^{\sim} \right] c^k \\
& - \sum_{r \in \mathcal{F}} \tilde{D}^{kr} C^{kr} \Sigma_{U0}^T M^r \ddot{q}^r - \sum_{r \in \mathcal{B}} \Phi^{kr} C^{kr} \dot{\omega}^r - \dot{h}^k \\
& - \tilde{\omega}^k h^k - \Sigma_{U0}^T \tilde{r}_k M^k \ddot{q}^k - \Sigma_{0U}^T M^k \ddot{q}^k \\
& + \mathfrak{M}_k \tilde{c}^k \sum_{r \in \mathcal{B} - \mathcal{F}} C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk} \\
& + \mathfrak{M} \sum_{r \in \mathcal{B} - \mathcal{F}} \tilde{D}^{kr} C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk} - \tilde{\omega}^k \Phi^{kk} \omega^k \\
& - \sum_{r \in \mathcal{B}} \mathfrak{M}_r \tilde{D}^{kr} C^{kr} \tilde{c}^r \dot{\omega}^r - \mathfrak{M}_k \tilde{c}^k \sum_{r \in \mathcal{B}} C^{kr} \tilde{D}^{rk} \dot{\omega}^r
\end{aligned} \tag{64}
$$

The appendage equation (Eq. 16) may be simplified as well (letting $R^k = 0$) to obtain

$$
\begin{aligned}
(k \in \mathcal{B}) \quad & M^k \left(U - \Sigma_{U0} \Sigma_{U0}^T \frac{M^k}{\mathfrak{M}} \right) \ddot{q}^k + K^k q^k \\
& = - M^k (\Sigma_{0U} - \tilde{r}_k \Sigma_{U0}) \dot{\omega}^k - M^k \Sigma_{U0} \sum_{r \in \mathcal{B}} C^{kr} \tilde{\dot{\omega}}^r D^{rk} \\
& \quad - M^k \Sigma_{U0} C^{k0} \frac{F}{\mathfrak{M}} + \lambda^k + M^k \Sigma_{U0} \sum_{r \in \mathcal{B} - k} C^{kr} \Sigma_{U0}^T \frac{M^r}{\mathfrak{M}} \ddot{q}^r \\
& \quad - M^k \Sigma_{U0} \sum_{r \in \mathcal{B} - \mathcal{F}} C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk}
\end{aligned} \tag{65}
$$

This appendage equation is analogous to that in Eq. (207) of Ref. 2, whose homogeneous solution has the form

$$q^k = \sum_{j=1}^{6n_k} a_j e^{\lambda_j t} \phi_k^j$$

where λ_j and ϕ_k^j are, respectively, eigenvalues and eigenvectors available from

$$(M' \lambda_j^2 + K') \phi_k^j = 0$$

and

$$M' = M^k \left(U - \Sigma_{U0} \Sigma_{U0}^T \frac{M^k}{\mathfrak{M}} \right)$$

$$K' = K^k$$

If ϕ_k is the $6n_k$ by $6n_k$ matrix

$$\phi_k \equiv [\phi_k^1 \ \phi_k^2 \cdots \phi_k^{6n_k}]$$

the transformation

$$q^k = \phi_k \eta^k \tag{66}$$

may be used to transform Eq. (65) into

$$\ddot{\eta}^k + \sigma_k^2 \eta^k = \phi_k^T L_k' \tag{67}$$

where

$$L_k' = -M^k(\Sigma_{0U} - \tilde{r}_k \Sigma_{U0})\dot{\omega}^k - M^k \Sigma_{U0} \sum_{r \in \mathfrak{B}} C^{kr} \tilde{\dot{\omega}}^r D^{rk}$$

$$- M^k \Sigma_{U0} C^{k0} \frac{F}{\mathfrak{M}} + \lambda^k + M^k \Sigma_{U0} \sum_{r \in \mathfrak{B} - k} C^{kr} \Sigma_{U0}^T \frac{M^r}{\mathfrak{M}} \phi_r \ddot{\eta}^r$$

$$- M^k \Sigma_{U0} \sum_{r \in \mathfrak{B} - \mathfrak{F}} C^{kr} \tilde{\omega}^r \tilde{\omega}^r D^{rk}$$

If the modal coordinates $\eta_1^k, \eta_2^k, \ldots, \eta_{6n_k}^k$ are now truncated to the set $\eta_1^k, \ldots, \eta_{N_k}^k$ (as symbolized by the overbar) and modal damping is also incorporated, Eq. (67) becomes

$$\ddot{\bar{\eta}}^k + 2\bar{\xi}_k \bar{\sigma}_k \dot{\bar{\eta}}^k + \bar{\sigma}_k^2 \bar{\eta}^k = \bar{\phi}_k^T L_k' \tag{68}$$

Returning to the vehicle substructure equation, Eq. (64), the truncated modal transformation, $q^k \approx \bar{\phi}_k \bar{\eta}^k$, may be substituted and the result combined with Eqs. (2), (3), (5), and (6) to give

$$A^{00}\dot{\omega}^0 + \sum_{j \in \mathfrak{P}} A^{0j} \ddot{\gamma}_j + \sum_{k \in \mathfrak{F}} A^{0k} \ddot{\bar{\eta}}^k = \sum_{k \in \mathfrak{B}} C^{0k} E^k \tag{69a}$$

$$(i \in \mathfrak{P}) \quad A^{i0}\dot{\omega}^0 + \sum_{j \in \mathfrak{P}} A^{ij} \ddot{\gamma}_j + \sum_{k \in \mathfrak{F}} A^{ik} \ddot{\bar{\eta}}^k = g^{i^T} \sum_{k \in \mathfrak{P}} \epsilon_{ik} C^{ik} E^k + \tau_i \tag{69b}$$

where

$$A^{00} = \sum_{k \in \mathfrak{B}} \sum_{r \in \mathfrak{B}} C^{0k} \Phi^{kr*} C^{r0}, \qquad \text{3 by 3}$$

$$A^{0j} = \sum_{k \in \mathfrak{B}} \sum_{r \in \mathfrak{P}} C^{0k} \Phi^{kr*} C^{rj} \epsilon_{jr} g^j, \qquad \text{3 by 1}$$

$$A^{0k} = C^{0k}\left(\bar{\Delta}^{k^T} + \sum_{r \in \mathfrak{B}} C^{kr} \tilde{D}^{rk} C^{rk} \bar{P}^k\right), \qquad \text{3 by } N_k$$

$$A^{i0} = g^{i^T} \sum_{k \in \mathfrak{P}} \sum_{r \in \mathfrak{B}} C^{ik} \epsilon_{ik} \Phi^{kr*} C^{r0}, \qquad \text{1 by 3}$$

$$A^{ij} = g^{i^T} \sum_{k \in \mathfrak{P}} \sum_{r \in \mathfrak{P}} C^{ik} \epsilon_{ik} \epsilon_{jr} \Phi^{kr*} C^{rj} g^j, \qquad \text{1 by 1}$$

$$A^{ik} = g^{i^T}\left(\epsilon_{ik}C^{ik}\bar{\Delta}^{k^T} + \sum_{r\in\mathcal{B}} \epsilon_{ir}C^{kr}\tilde{D}^{rk}C^{rk}\bar{P}^k\right), \qquad 1 \text{ by } N_k$$

$$E^k = T^k - \tau_R^k - \tilde{\omega}^k \mathcal{J}^k(\tilde{\omega}^k + \dot{\psi}_R^k) + \sum_{r\in\mathcal{B}} \tilde{D}^{kr}C^{kr}F^r$$

$$+ \left[\tilde{F}^k - \left(C^{k0}\frac{\mathfrak{M}_k}{\mathfrak{M}}F\right)^{\sim}\right]c^k + \mathfrak{M}\sum_{r\in\mathcal{B}-\mathcal{F}} \tilde{D}^{kr}C^{kr}\tilde{\omega}^r\tilde{\omega}^r D^{rk}$$

$$- \tilde{\omega}^k\Phi^{kk}\omega^k - \sum_{r\in\mathcal{B}}\Phi^{kr*}\sum_{j\in\mathcal{P}} C^{rj}\epsilon_{jr}\tilde{\omega}^j g^j\dot{\gamma}_j$$

$$+ \mathfrak{M}_k\tilde{c}^k\sum_{r\in\mathcal{B}-\mathcal{F}} C^{kr}\tilde{\omega}^r\tilde{\omega}^r D^{rk}, \qquad 3 \text{ by } 1$$

$$\Phi^{kr*} = \bar{\Phi}^{kr}C^{kr} + \mathfrak{M}_k\tilde{c}^k C^{kr}\tilde{D}^{rk} + \mathfrak{M}_r\tilde{D}^{kr}C^{kr}\tilde{c}^r, \qquad 3 \text{ by } 3$$

$$\bar{\Delta}^k = \bar{\phi}_k^T M^k(\Sigma_{0U} - \tilde{r}_k\Sigma_{U0}), \qquad N_k \text{ by } 3$$

$$\bar{P}^k = \Sigma_{U0}^T M^k\bar{\phi}^k, \qquad 3 \text{ by } N_k$$

($\bar{\Phi}^{kr}$ does not include the effects of appendage deformation.)

As in Eqs. (32) and (33), substitutions have been made for $\dot{h}^k$ and h^k based on restriction to three orthogonal axisymmetric rotors in $\mathscr{s}_k$, with spin axes aligned to the unit vectors $\{\mathbf{b}^k\}$, and the relations in Eqs. (43)–(45). Again, it is to be understood that any rotor's moments of inertia are to be included in $\mathbf{J}^k$, the undeformed substructure's inertia dyadic for o_k, and its mass is included in the substructure mass, $\mathfrak{M}_k$.

Operating on the appendage equation, Eq. (68), in a similar way provides

$$(k \in \mathcal{F}) \qquad A^{k0}\dot{\omega}^0 + \sum_{j\in\mathcal{P}} A^{kj}\ddot{\gamma}_j + \sum_{r\in\mathcal{F}} A^{rk}\ddot{\eta}^r = Q^k \qquad (70)$$

where

$$A^{k0} = \bar{\Delta}^k C^{k0} - \bar{P}^{k^T}\sum_{r\in\mathcal{B}} C^{kr}\tilde{D}^{rk}C^{r0}, \qquad N_k \text{ by } 3$$

$$A^{kj} = \left(\bar{\Delta}^k C^{kj}\epsilon_{jk} - \bar{P}^{k^T}\sum_{r\in\mathcal{B}} C^{kr}\tilde{D}^{rk}C^{rj}\epsilon_{jr}\right)g^j, \qquad N_k \text{ by } 1$$

$$A^{rk} = -\bar{P}^{k^T}C^{kr}\frac{\bar{P}^r}{\mathfrak{M}}, \quad (r \neq k); \qquad N_k \text{ by } N_r$$

$$A^{rk} = U, \quad (r = k); \qquad N_k \text{ by } N_k$$

$$Q^k = -2\bar{\xi}_k\bar{\sigma}_k\dot{\bar{\eta}}^k - \bar{\sigma}_k^2\bar{\eta}^k - \bar{P}^{k^T}C^{k0}\frac{F}{\mathfrak{M}} + \bar{\phi}_k^T\lambda^k$$

$$- \sum_{j\in\mathcal{P}}\left(\tilde{\bar{\Delta}}^k C^{kj}\epsilon_{jk} - \bar{P}^{k^T}\sum_{r\in\mathcal{B}} C^{kr}\tilde{D}^{rk}C^{rj}\epsilon_{jr}\right)\tilde{\omega}^j g^j\dot{\gamma}_j$$

$$- \bar{P}^{k^T}\sum_{r\in\mathcal{B}-\mathcal{F}} C^{kr}\tilde{\omega}^r\tilde{\omega}^r D^{rk}, \qquad N_k \text{ by } 1$$

where modal damping, $\bar{\xi}_k$, has been added (see discussion in Section IIIA).

The substructure and appendage equations may now be combined into a single matrix equation of the form $A\dot{x} = B$,

$$\begin{bmatrix} A^{00} & A^{0j} & A^{0k} \\ A^{i0} & A^{ij} & A^{ik} \\ A^{k0} & A^{kj} & A^{rk} \end{bmatrix} \begin{bmatrix} \dot{\omega}^0 \\ \ddot{\gamma}_j \\ \ddot{\bar{\eta}}^k \end{bmatrix} = \begin{bmatrix} \sum_{k\in\mathcal{B}} C^{0k}E^k \\ g^{i^T}\sum_{k\in\mathcal{P}}\epsilon_{ik}C^{ik}E^k + \tau_i \\ Q^k \end{bmatrix} \tag{71}$$

Again the elements of A are, in general, time-variable because of substructure relative motion. A is also symmetric.

Very often, one can justify making the assumption that *all* the variables, i.e., ω^0, γ_j, $\bar{\eta}^k$, and their derivatives are in some sense "small" and a complete linearization of Eq. (71) may be carried out. The computational benefits of a total linearization are quite substantial since the coefficient matrix, A, then becomes formally constant, allowing its inverse to be computed only once, in advance of numerical integration.

If each symbol in Eq. (71) is expanded into three parts, the first being free of the variables ω^0, γ_j, $\bar{\eta}^k$, and their derivatives (indicated by overbar), the second being linear in these variables (indicated by overcaret), and the third containing terms above the first degree in the variables (indicated by three dots), and if one then determines explicit expressions for the new barred and careted symbols from their definitions, the linearized form of Eq. (71) becomes

$$\bar{A}^{00}\dot{\omega}^0 + \sum_{j\in\mathcal{P}}\bar{A}^{0j}\ddot{\gamma}_j + \sum_{k\in\mathcal{F}}\bar{A}^{0k}\ddot{\bar{\eta}}^k = \sum_{k\in\mathcal{B}}\left[\bar{C}^{0k}(\bar{E}^k + \hat{E}^k) + \hat{C}^{0k}\bar{E}^k\right] \tag{72a}$$

$$(i\in\mathcal{P}) \qquad \bar{A}^{i0}\dot{\omega}^0 + \sum_{j\in\mathcal{P}}\bar{A}^{ij}\ddot{\gamma}_j + \sum_{k\in\mathcal{F}}\bar{A}^{ik}\ddot{\bar{\eta}}^k$$

$$= g^{i^T}\sum_{k\in\mathcal{P}}\epsilon_{ik}\left[\bar{C}^{ik}(\bar{E}^k + \hat{E}^k) + \hat{C}^{ik}\bar{E}^k\right] + \bar{\tau}_i + \hat{\tau}_i \tag{72b}$$

$$(k\in\mathcal{F}) \qquad \bar{A}^{k0}\dot{\omega}^0 + \sum_{j\in\mathcal{P}}\bar{A}^{kj}\ddot{\gamma}_j + \sum_{r\in\mathcal{F}}\bar{A}^{rk}\ddot{\bar{\eta}}^r = \bar{Q}^k + \hat{Q}^k \tag{72c}$$

where

$$C^{i0} = \overline{C}^{i0} + \hat{C}^{i0} + \cdots$$

$$\tau_i = \overline{\tau}_i + \hat{\tau}_i + \cdots$$

$$A^{kj} = \overline{A}^{kj} + \hat{A}^{kj} + \cdots$$

$$E^k = \overline{E}^k + \hat{E}^k + \cdots$$

etc.,

and

$$\overline{C}^{rj} = \overline{C}^{jr} = U = 3 \text{ by 3 identity matrix}$$

$$\hat{C}^{rj} = -\gamma_r \tilde{g}_r, \qquad (r > j)$$

$$\hat{C}^{jr} = \gamma_r \tilde{g}_r = (\hat{C}^{rj})^T$$

Specifically,

$$\overline{A}^{00} = \sum_{k \in \mathfrak{B}} \sum_{r \in \mathfrak{B}} \overline{\Phi}^{kr}$$

$$\overline{A}^{0j} = \sum_{k \in \mathfrak{B}} \sum_{r \in \mathfrak{P}} \overline{\Phi}^{kr} \epsilon_{jr} g^j$$

$$\overline{A}^{0k} = \overline{\Delta}^{k^T} + \sum_{r \in \mathfrak{B}} \tilde{D}^{rk} \overline{P}^k$$

$$\overline{A}^{i0} = g^{i^T} \sum_{k \in \mathfrak{P}} \sum_{r \in \mathfrak{B}} \epsilon_{ik} \overline{\Phi}^{kr}$$

$$\overline{A}^{ij} = g^{i^T} \sum_{k \in \mathfrak{P}} \sum_{r \in \mathfrak{P}} \epsilon_{ik} \epsilon_{jr} \overline{\Phi}^{kr} g^j$$

$$\overline{A}^{ik} = g^{i^T} \left(\epsilon_{ik} \overline{\Delta}^{k^T} + \sum_{r \in \mathfrak{B}} \epsilon_{ir} \tilde{D}^{rk} \overline{P}^k \right)$$

$$\overline{E}^k = \overline{T}^k - \overline{\tau}_R^k + \sum_{r \in \mathfrak{B}} \tilde{D}^{kr} \overline{F}^r$$

$$\hat{E}^k = \hat{T}^k - \hat{\tau}_R^k - \tilde{\omega}^k \mathfrak{g}^k \dot{\psi}_R^k + \sum_{r \in \mathfrak{B}} \tilde{D}^{kr} \hat{C}^{kr} \overline{F}^r + \sum_{r \in \mathfrak{B}} \tilde{D}^{kr} \hat{F}^r + \left[\bar{\tilde{F}}^k - \left(\frac{\mathfrak{M}_k}{\mathfrak{M}} \overline{F} \right)^{\tilde{}} \right] c^k$$

$$\overline{A}^{k0} = \overline{\Delta}^k - \overline{P}^{k^T} \sum_{r \in \mathfrak{B}} \tilde{D}^{rk}$$

$$\bar{A}^{kj} = \left(\bar{\Delta}^k \epsilon_{jk} - \bar{P}^{k^T} \sum_{r \in \mathcal{B}} \tilde{D}^{rk} \epsilon_{jr}\right) g^j$$

$$\bar{A}^{rk} = -\bar{P}^{k^T} \frac{\bar{P}^r}{\mathfrak{M}}, \quad (r \neq k)$$

$$\bar{A}^{rk} = U, \qquad (r = k)$$

$$\bar{Q}^k = -\bar{P}^{k^T} \frac{\bar{F}}{\mathfrak{M}} + \bar{\Phi}_k^T \bar{\lambda}^k$$

$$\hat{Q}^k = -2\bar{\xi}_k \bar{\sigma}_k \dot{\hat{\eta}}^k - \bar{\sigma}_k^2 \hat{\eta}^k - \bar{P}^{k^T} \frac{\hat{C}^{k0}\bar{F} + \hat{F}}{\mathfrak{M}} + \bar{\phi}_k^T \hat{\lambda}^k$$

It would remain then to determine $\bar{T}^k$, $\hat{T}^k$, $\bar{F}^k$, $\hat{F}^k$, $\bar{F}$, $\hat{F}$, $\bar{\lambda}^k$, $\hat{\lambda}^k$, $\bar{\tau}_i$, $\hat{\tau}_i$, etc., for the particular system under study and to carry out the computations in Eq. (72). However, in constructing a subroutine to perform these computations, it was found to be more efficient to directly manipulate the combined form

$$\bar{A}^{00}\dot{\omega}^0 + \sum_{j \in \mathcal{P}} \bar{A}^{0j} \ddot{\gamma}_j + \sum_{k \in \mathcal{F}} \bar{A}^{0k} \ddot{\hat{\eta}}^k = \sum_{k \in \mathcal{B}} \hat{\bar{C}}^{0k} \hat{\bar{E}}^k \tag{73a}$$

$$(i \in \mathcal{P}) \quad \bar{A}^{i0}\dot{\omega}^0 + \sum_{j \in \mathcal{P}} \bar{A}^{ij} \ddot{\gamma}_j + \sum_{k \in \mathcal{F}} \bar{A}^{ik} \ddot{\hat{\eta}}^k = g^{i^T} \sum_{k \in \mathcal{P}} \epsilon_{ik} \hat{\bar{C}}^{ik} \hat{\bar{E}}^k + \hat{\bar{\tau}}_i \tag{73b}$$

$$(k \in \mathcal{F}) \quad \bar{A}^{k0}\dot{\omega}^0 + \sum_{j \in \mathcal{P}} \bar{A}^{kj} \ddot{\gamma}_j + \sum_{r \in \mathcal{F}} \bar{A}^{rk} \ddot{\hat{\eta}}^r = \hat{\bar{Q}}^k$$

where

$$\hat{\bar{E}}^k = \bar{E}^k + \hat{E}^k$$

$$\hat{\bar{C}}^{ik} = \bar{C}^{ik} + \hat{C}^{ik}$$

$$\hat{\bar{\tau}}_i = \bar{\tau}_i + \hat{\tau}_i$$

etc.

By avoiding the separation into the parts $\bar{E}^k$, $\hat{E}^k$, etc., the computation becomes more efficient even though some second-order terms in the linearized variables are retained.

B. Subroutines MBDYFN, MBDYFL

The Fortran V subroutines MBDYFN and MBDYFL were written to provide the solutions to Eqs. (71) and (73), respectively. As in the case of MBDYFR, these routines are also exercised by either of two call statements, the first of which initializes the program with the system constants.

Initializing Call Statements

CALL MBDYFN(NC, H, MB, MS, PB, PS, G, PI,
NF, F, EIG, REC, RF, WF, ZF)

or

CALL MBDYFL(NC, H, MB, MS, PB, PS, G, PI,
NF, F, EIG, REC, RF, WF, ZF)

All the arguments in these call statements are defined exactly as given in IIIB, with the exception of the two new arguments, EIG and REC. Notice that the MBDYFR inputs ER, EI, SR, and MF no longer are used in these routines. The input arrays RF and EIG are used by the subroutine *only* if there are nonzero external forces and torques λ^k applied to an appendage.

EIG(n, i, j) = array of elements of $\bar{\phi}^j_k$; $n = 1, 2, \ldots, n_f$; $i = 1, 2, \ldots, 6n_k$; $k = F(n, 1)$; $j = 1, 2, \ldots, N_k$. (Note! This array is not used by the routine if λ^k, for all $k \in \mathfrak{F}$, is zero.)

REC(n, i, j) = array containing the "rigid-elastic coupling coefficients," $\bar{\Delta}^k$ and $\bar{P}^k$; $n = 1, 2, \ldots, n_f$; $i = 1, 2, \ldots, 6$; $k = F(n, 1)$; $j = 1, 2, \ldots, N_k$. (For $i = 1, 2, 3$, the elements of REC are those of $\bar{P}^k$; for $i = 4, 5, 6$, the elements are those of $\bar{\Delta}^{k^T}$.)

In order to compute the angular accelerations $\dot{\omega}^0, \ddot{\gamma}_1, \ldots, \ddot{\gamma}_n$, and the modal coordinate acceleration vectors $\ddot{\bar{\eta}}^k$ ($k \in \mathfrak{F}$) at every numerical integration step, the simulation must repeatedly enter the subroutine using the dynamic call statement.

Dynamic Call Statement

CALL MRATE(NC, TH, TB, TS, FB, FS, TF, FF, GM,
GMD, GMDD, ET, ETD, WO, WDOT, ETDD, HM)

where

ET(n, i) = array of appendage modal coordinates, $\bar{\eta}^k$; $n = 1, \ldots, n_f$; $k = F(n, 1)$, $i = 1, \ldots, N_k$.

ETD(n, i) = array of modal coordinate rates, $\dot{\bar{\eta}}^k$; $n = 1, \ldots, n_f$; $k = F(n, 1)$, $i = 1, \ldots, N_k$.

ETDD(n, i) = solution array for modal coordinate accelerations, $\ddot{\bar{\eta}}^k$; $n = 1, \ldots, n_f$; $k = F(n, 1)$, $i = 1, \ldots, N_k$.

and all other arguments are defined exactly as in IIIB.

Again, it should be noted that MBDYFN and MBDYFL do not incorporate the terms in E^k that describe rotor torques on $\mathscr{A}_k$. The user must include these terms, if rotors are present, in T^k (or $\hat{\bar{T}}^k$) as it is formed in the main program.

Also, if any of the γ_i are to be *prescribed*, appropriate values of $\ddot{\gamma}_i$, as well as γ_i and $\dot{\gamma}_i$, must be supplied to the subroutine by way of the MRATE dummy arguments GMDD, GM, and GMD, respectively.

When either the MBDYFN or the MBDYFL subroutine is used, the main calling program must contain Fortran "type" and storage allocation statements. The mandatory statements are:

Required Specification Statements

INTEGER NC, NF, H(n_c, 2), F(n_f, 3), PI(n + 1)

REAL MB(7), MS(n_c, 7), PB(n_c, 3), PS(n_c, n_c, 3),
G(n, 3), TH(n), TB(3), TS(n_c, 3), FB(3), FS(n_c, 3),
GM(n), GMD(n), GMDD(n), EIG(n_f, 6n_k, N_k), REC(n_f, 6, N_k),
RF(n_f, n_k, 3), WF(n_f, N_k), ZF(n_f, N_k),
TF(n_f, n_k, 3), FF(n_f, n_k, 3), ET(n_f, N_k),
ETD(n_f, N_k), WO(3)

DOUBLE PRECISION WDOT(n + 3), ETDD(n_f, N_k)

In order that storage allocation for arrays internal to MBDYFN and MBDYFL be minimized, the following statement must appear in the subroutine:

PARAMETER QH = n, QC = n_c, QF = n_f, NK = n_k, NKT = N_k

The proper placement of this statement in MBDYFN and MBDYFL is shown in their listing (Appendices D and E).

C. Sample Problems

To illustrate the use of subroutines MBDYFN and MBDYFL, a sample problem suitable for computer simulation will be described. The test vehicle to be simulated has the configuration shown in Fig. 10—a rigid central body, b_0, a rigid platform, b_1, which is hinged to b_0 (2 degrees of freedom), and a flexible appendage, a_0, also attached to b_0.

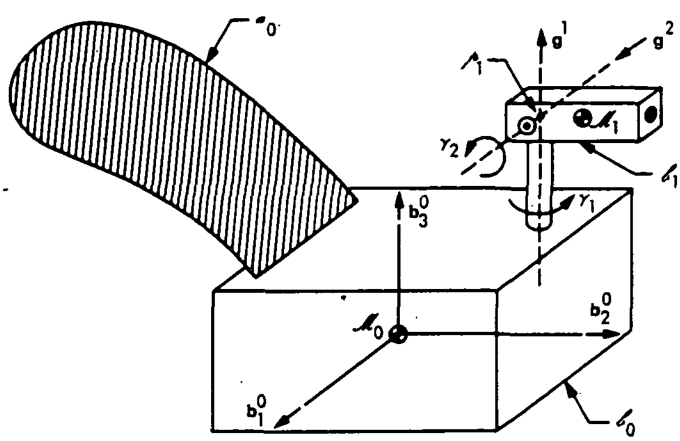

Fig. 10. MBDYFN, MBDYFL simulation test vehicle

For this example, the numbers used to describe the test vehicle's mass properties, including the appendage, were taken from an actual spacecraft design. The appendage model includes the characteristic vibration modes of four solar panels, a parabolic antenna, and several other structural members.

Test Vehicle Constants

The following numerical constants are required for initializing the subroutines:

$$\mathfrak{M}_0 = 79.0 \text{ kg}$$

$$\mathfrak{M}_1 = 1.93 \text{ kg}$$

$$\bar{J}^0 = \begin{bmatrix} 1230. & 16.29 & 43.45 \\ & 1290. & -61.75 \\ \text{sym.} & & 1650. \end{bmatrix} \text{kg-m}^2$$

$$\bar{J}^1 = \begin{bmatrix} 4.75 & 0. & 0. \\ & 5.53 & 0. \\ \text{sym.} & & 1.32 \end{bmatrix} \text{kg-m}^2$$

Let the modal model for appendage a_0 ($\mathcal{A}_1$) be truncated to seven modes, i.e., $N_0 = 7$. Thus,

$$\bar{P}^0 = \Sigma_{U0}^T M^0 \bar{\phi}_0 =$$

$$\left[\begin{array}{c|c|c|c|c|c|c} .0338 & .0106 & .0023 & .0032 & -.6055 & -.3050 & -.0276 \\ .0017 & .0011 & -.0182 & .0010 & -.5381 & 1.753 & .1051 \\ -.8678 & -.00005 & 0. & 2.234 & 1.962 & .5585 & .3919 \end{array}\right]$$

kg-m

$$\bar{\Delta}^{0^T} = (\Sigma_{0U}^T + \Sigma_{U0}^T \tilde{r}_0) M^0 \bar{\phi}_0 =$$

$$\left[\begin{array}{c|c|c|c|c|c|c} .0814 & .4236 & 21.30 & -.4081 & 7.577 & -4.320 & 2.032 \\ 17.17 & 12.30 & -.2386 & 5.930 & .4020 & -.1589 & -2.061 \\ .0080 & .0019 & .0009 & -.0521 & 2.520 & -.9205 & .2761 \end{array}\right]$$

kg-m^2

$$\bar{\sigma}_0 = 2\pi[.5756 \quad .6134 \quad .6134 \quad .6307 \quad 2.723 \quad 2.963 \quad 3.047]^T \text{ rad/s}$$

$$\bar{\xi}_0 = [.20 \quad .20 \quad .20 \quad .20 \quad .05 \quad .05 \quad .01]^T$$

Also, let

$$g^1 = [0. \quad 0. \quad 1.]^T$$

$$g^2 = [1. \quad 0. \quad 0.]^T$$

$$p^{01} = [0. \quad 0. \quad 0.]^T, \quad p^{10} = [0. \quad 0. \quad 0.]^T$$

Since no external forces or torques will be applied to the appendage, the eigenvector matrix $\bar{\phi}_0$ is not needed, nor is the matrix $\tilde{r}_0$. Finally,

$$n_c = 1, \quad n_f = 1, \quad n_0 = 1, \quad N_0 = 7$$

$$h_1 = 0, \quad d_1 = 2, \quad n = 2$$

The integer n_0, which indicates the number of sub-bodies in the appendage model and is only required if external forces and torques are applied to appendage α_0, has been set to the smallest acceptable value that satisfies dimensioning requirements.

The initializing call statement arguments therefore become

NC = 1

H = [0 2]

MB = [1230. 1290. 1650. −16.29 −43.45 61.75 79.0]

MS = [4.75 5.53 1.32 0. 0. 0. 1.93]

PB = 0

PS = 0

$$G = \begin{bmatrix} 0. & 0. & 1. \\ 1. & 0. & 0. \end{bmatrix}$$

PI = [0 0 1] (assuming no prescribed hinge motions)

NF = 1

F = [0 1 7]

EIG = 0

REC =

.0338	.0106	.0023	.0032	−.6055	−.3050	−.0276
.0017	.0011	−.0182	.0010	−.5381	1.753	.1051
−.8678	−.00005	0.	2.234	1.962	.5585	.3919
.0814	.4236	21.30	−.4081	7.577	−4.320	2.032
17.17	12.30	−.2386	5.930	.4020	−.1589	−2.061
.0080	.0019	.0009	−.0521	2.520	−.9205	.2761

$$WF = 2\pi[.5756 \quad .6134 \quad .6134 \quad .6307 \quad 2.723 \quad 2.963 \quad 3.047]^T$$

$$ZF = [.20 \quad .20 \quad .20 \quad .20 \quad .05 \quad .05 \quad .01]$$

Test Vehicle Dynamics

As before, the platform hinge connections will be defined as being of the linear spring and viscous damper type, but the position commands will be deleted, so that

$$\tau_1 = -K_1\gamma_1 - B_1\dot{\gamma}_1$$

$$\tau_2 = -K_2\gamma_2 - B_2\dot{\gamma}_2$$

where

$$K_1 = 900. \text{ n-m/rad}$$

$$K_2 = 850. \text{ n-m/rad}$$

$$B_1 = 100. \text{ n-m-s/rad}$$

$$B_2 = 100. \text{ n-m-s/rad}$$

The vehicle response to be simulated in this example will be that due to an arbitrary sequence of force and torque pulses applied to the reference body, $\mathscr{b}_0$. A rectangular pulse of thrust will be applied in the $\mathbf{b}_3^0$ direction with magnitude 300 n and a duration of 2 s, starting at $t = .5$ s. This will be followed by a 1-s torque pulse in the $\mathbf{b}_1^0$ direction of magnitude 10. n-m, starting at $t = 3.5$ s. And the last disturbance will be a 1-s torque pulse in the $\mathbf{b}_2^0$ direction of magnitude 10. n-m, starting at $t = 6.5$ s. The computer program for this dynamic simulation is given in Fig. 11.

Initially, the system is assumed to be completely at rest. Again, the CSSL III language function, "STEP," is used to construct the applied pulses. Only the angular rates of $\mathscr{b}_0$ are calculated in this example; its inertial angular position is not computed. Appendage modal coordinate rates and positions are both provided, although only the rates are plotted in the system responses of Fig. 12. A sample of the printed output is shown in Fig. 13.

Notice that by far the greatest disturbing effect to both platform and flexible appendage is due to the applied force. However, the changes in ω^0 magnitude due to the torque disturbances are quite significant. It is not clear to what extent the platform vibrations are coupling with appendage vibrations and reference body motion, although the platform rotations are small in magnitude.

It is apparent that the applied force (fixed with respect to $\mathscr{b}_0$) caused some slight accumulation of system angular momentum as the system mass center moved in response to platform and appendage vibrations. This small amount (.17 n-m-s) was dwarfed, however, by the next pulse of torque, so that after 4.5 s, the angular momentum should have been approximately 10 n-m-s. The last torque pulse, applied *orthogonally* to the preceding one, would then raise the total angular momentum magnitude to slightly more than $\sqrt{(10)^2 + (10)^2} = 14.14$ n-m-s. The simulation printout shows a computed value of 14.25 n-m-s.

```
CSSL III  JET PROPULSION LABORATORY  040374-A002H  021775-021424

... START                    T(RUN)=  18.518 T(TASK)=     .003  CTP    =    .544
                                              DT(TASK)=    .003 DCTP   =    .544

     PROGRAM    2-BODY VEHICLE WITH FLEX. APPENDAGE
     *SC4020    BLDG/198,BOX/601,CAMERA/91N,FRAMES/50
     COMMENT
           ARRAY MB(7),MS(1,7),PB(3),PS(1,1,3),G(2,3)
           ARRAY EIG(1,6,7),RF(1,1,3),REC(1,6,7),WF(1,7),ZF(1,7)
           ARRAY TB(3),TS(1,3),FB(3),FS(1,3),GM(2),GMD(2),GMDD(2)
           ARRAY TM(2),WO(3),TF(1,1,3),FF(1,1,3),ET(1,7),ETD(1,7)
           DOUBLE PRECISION WDOT(5),ETDD(1,7)
           INTEGER NC,NF,H(1,2),F(1,3),PI(3),L
     DATA H(1,1)/0/H(1,2)/2/PI/0,0,1/
     DATA F(1,1)/0/F(1,2)/1/F(1,3)/7/
     DATA MB/1230.,1290.,1650.,-16.29,-43.45,61.75,79.0/
     DATA MS/9.75,5.53,1.32,0.,0.,0.,1.93/
     DATA G(1,3)/1./G(2,1)/1./
     DATA REC/.03375,.001654,-.8678,.08135,17.17,.007955,...
              .01055,.001104,-.4608E-4,.4236,12.3,.001859,...
              .002335,-.01818,-.4731E-5,21.3,-.2386,.000918,...
              .003244,.001014,2.234,-.4081,5.930,-.05211,...
              -.6055,-.5381,1.962,7.577,.4020,2.520,...
              -.3050,1.753,.5585,-4.32,-.1589,-.9205,...
              -.02762,.1051,.3919,2.032,-2.061,.2761/
     DATA WF/.5756,.61337,.61337,.63071,2.723,2.963,3.047/
     DATA ZF/.20,.20,.20,.20,.05,.05,.01/
           CONSTANT FINTIM=10.,CLKTIM=900.,PIE=3.14159265
           CONSTANT K1=900.,B1=100.,K2=850.,B2=100.
     INITIAL
           NC=1   $   NF=1
           DO 57 L=1,7
     57..  WF(1,L)=WF(1,L)*2.*PIE
           CALL MBDYFN(NC,H,MB,MS,PB,PS,G,PI,NF,F,EIG,REC,RF,WF,Z,)
     END
     DYNAMIC
           IF(TIME.GT.FINTIM) GO TO FIN
           STPCLK   CLKTIM
           OUTPUT 10,W1,W2,W3,NX,NY,FZ,ETA1,ETA2,ETA3,ETA4,ETA5,ETA6,ETA7,...
                      ETD1,ETD2,ETD3,ETD4,ETD5,ETD6,ETD7,ANGM,W1D,W2D,W3D,...
                      GM1,GM1D,GM2,GM2D
           PREPAR W1,W2,W3,NX,NY,FZ,ETD1,ETD2,ETD3,ETD4,ETD5,ETD6,ETD7,...
                      ANGM,GM1,GM2,GM1D,GM2D
     DERIVATIVE   BODY2F
           VARIABLE TIME=0.  $  CINTERVAL  CI=.01
           XERROR W1=1.E-6     $    MERROR W1=1.E-6
     NOSORT
           GMD(1)=GM1D   $   GM(1)=GM1
           GMD(2)=GM2D   $   GM(2)=GM2
           ET(1,1)=ETA1 $ ET(1,2)=ETA2 $ ET(1,3)=ETA3 $ ET(1,4)=ETA4
           ET(1,5)=ETA5 $ ET(1,6)=ETA6 $ ET(1,7)=ETA7
           ETD(1,1)=ETD1 $ ETD(1,2)=ETD2 $ ETD(1,3)=ETD3 $ ETD(1,4)=ETD4
           ETD(1,5)=ETD5 $ ETD(1,6)=ETD6 $ ETD(1,7)=ETD7
           WO(1)=W1  $  WO(2)=W2  $  WO(3)=W3  $  ANGM=HM
     COMMENT...
                  HINGE TORQUES
```

Fig. 11. Simulation program for test vehicle dynamics using MBDYFN

```
COMMENT
       TH(1)=-K1*GM1 - B1*GM1D
       TH(2)=-K2*GM2 - B2*GM2D
COMMENT...
                 FORCE EQUATION
COMMENT
       FZ=(STEP(.5,TIME)-STEP(2.5,TIME))*300.
       FB(3)=FZ
COMMENT...
                 ENGINE TORQUE
COMMENT
       NX=(STEP(3.5,TIME)-STEP(4.5,TIME))*10.
       NY=(STEP(6.5,TIME)-STEP(7.5,TIME))*10.
       TB(1)=NX   $   TB(2)=NY
COMMENT...
                 SOLUTION FOR SYSTEM ACCELERATIONS
COMMENT
       CALL MRATE(NC,TH,TB,TS,FB,FS,TF,FF,GM,GMD,GMDD,ET,ETD,W0,WDOT,...,
       ETDD,HM)
       W1D=WDOT(1)   $   W2D=WDOT(2)   $   W3D=WDOT(3)
COMMENT...
                 SYSTEM RATES AND POSITIONS
COMMENT
       W1=INTEG(WDOT(1),0.)
       W2=INTEG(WDOT(2),0.)
       W3=INTEG(WDOT(3),0.)
       ETD1=INTEG(ETDD(1,1),0.)   $   ETA1=INTEG(ETD1,0.)
       ETD2=INTEG(ETDD(1,2),0.)   $   ETA2=INTEG(ETD2,0.)
       ETD3=INTEG(ETDD(1,3),0.)   $   ETA3=INTEG(ETD3,0.)
       ETD4=INTEG(ETDD(1,4),0.)   $   ETA4=INTEG(ETD4,0.)
       ETD5=INTEG(ETDD(1,5),0.)   $   ETA5=INTEG(ETD5,0.)
       ETD6=INTEG(ETDD(1,6),0.)   $   ETA6=INTEG(ETD6,0.)
       ETD7=INTEG(ETDD(1,7),0.)   $   ETA7=INTEG(ETD7,0.)
       GM1D=INTEG(WDOT(4),0.)   $   GM1=INTEG(GM1D,0.)
       GM2D=INTEG(WDOT(5),0.)   $   GM2=INTEG(GM2D,0.)
END
END
END
TERMINAL
FIN.. CONTINUE
END
END
```

Fig. 11 (contd)

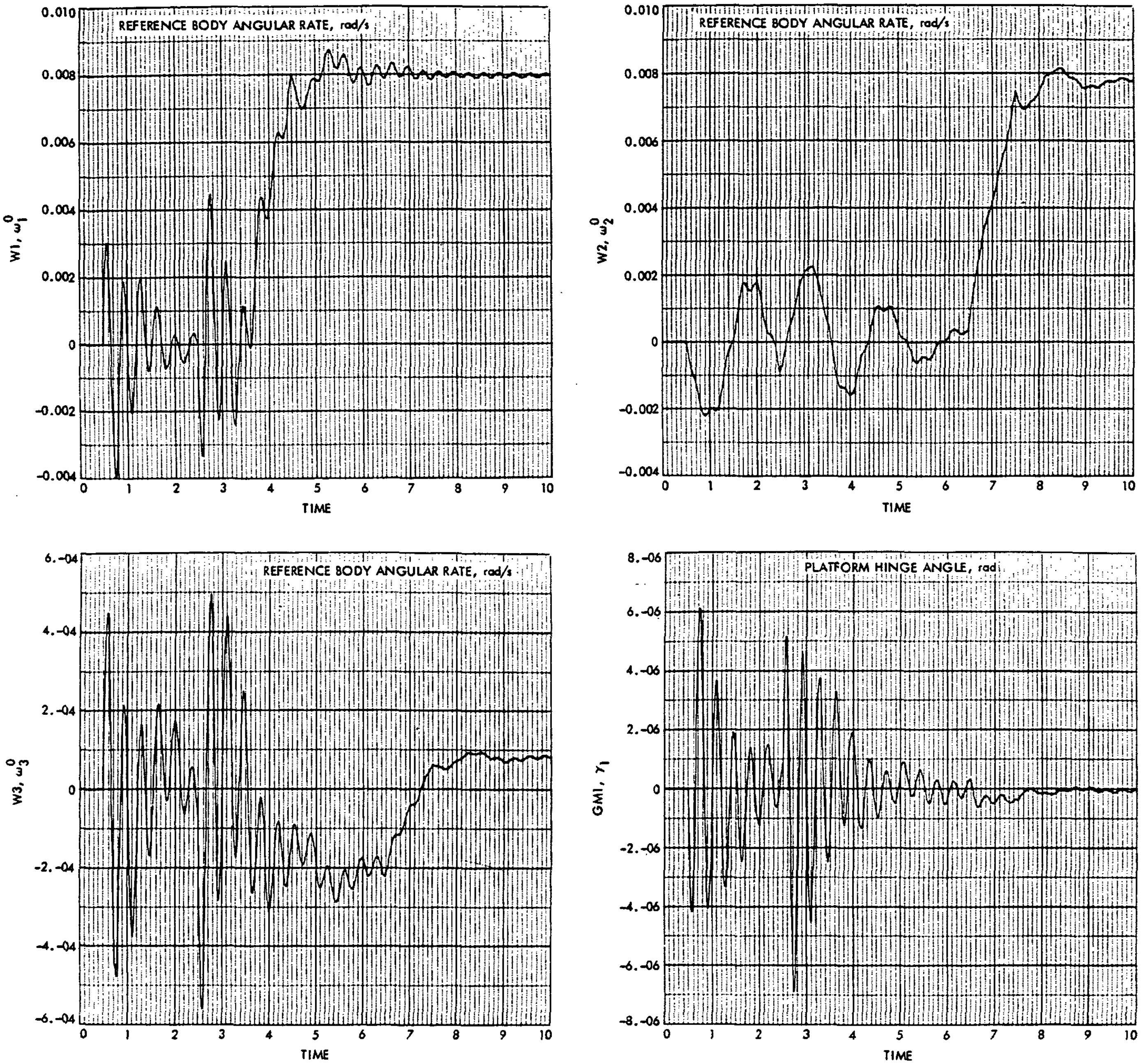

Fig. 12. Test vehicle simulation results using MBDYFN

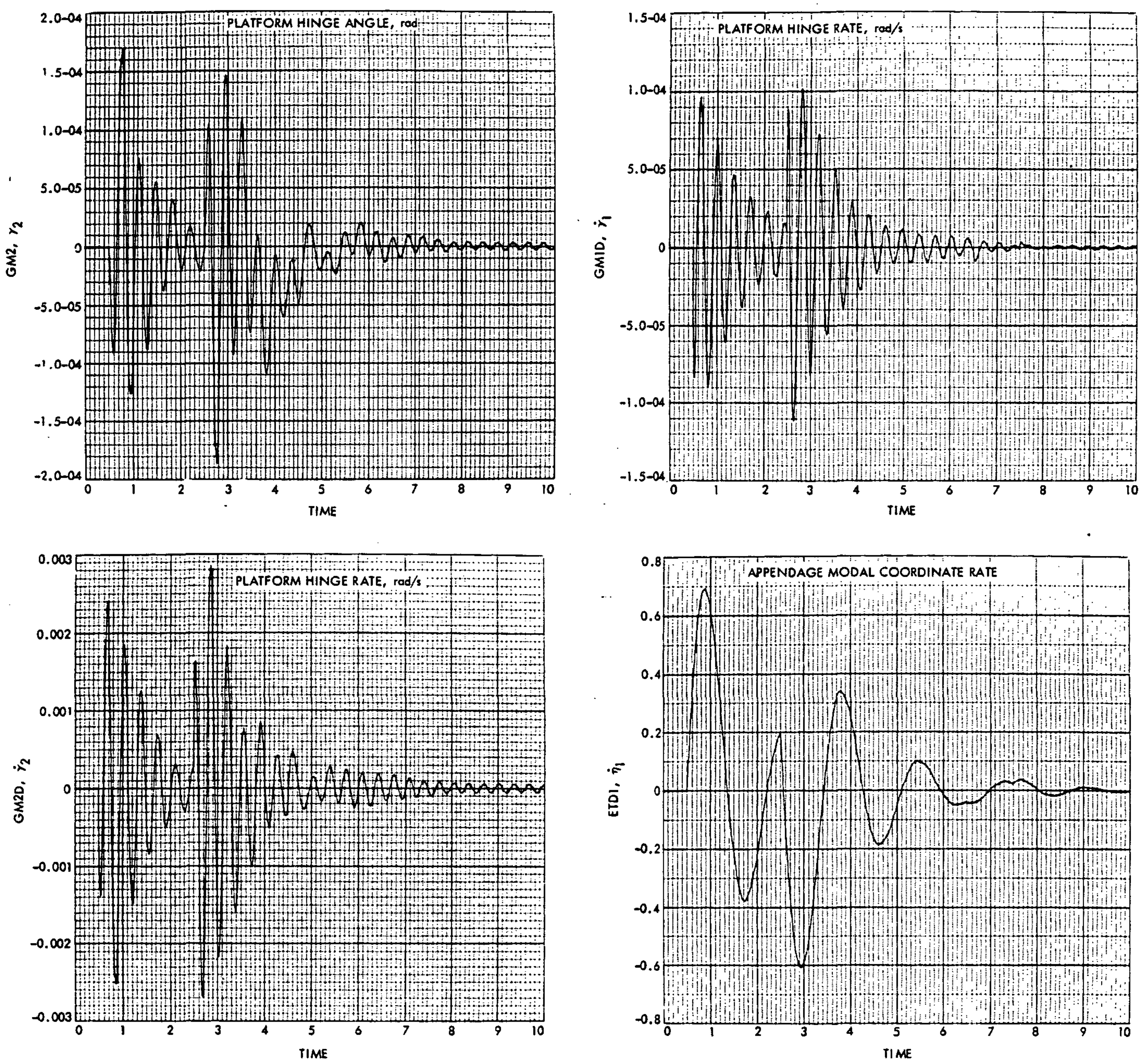
PLATFORM HINGE ANGLE, rad
GM2, γ_2
2.0-04
1.5-04
1.0-04
5.0-05
0
-5.0-05
-1.0-04
-1.5-04
-2.0-04
0 1 2 3 4 5 6 7 8 9 10
TIME
PLATFORM HINGE RATE, rad/s
GM1D, $\dot{\gamma}_1$
1.5-04
1.0-04
5.0-05
0
-5.0-05
-1.0-04
-1.5-04
0 1 2 3 4 5 6 7 8 9 10
TIME
PLATFORM HINGE RATE, rad/s
GM2D, $\dot{\gamma}_2$
0.003
0.002
0.001
0
-0.001
-0.002
-0.003
0 1 2 3 4 5 6 7 8 9 10
TIME
APPENDAGE MODAL COORDINATE RATE
ETD1, $\dot{\eta}_1$
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
0 1 2 3 4 5 6 7 8 9 10
TIME

Fig. 12 (contd)

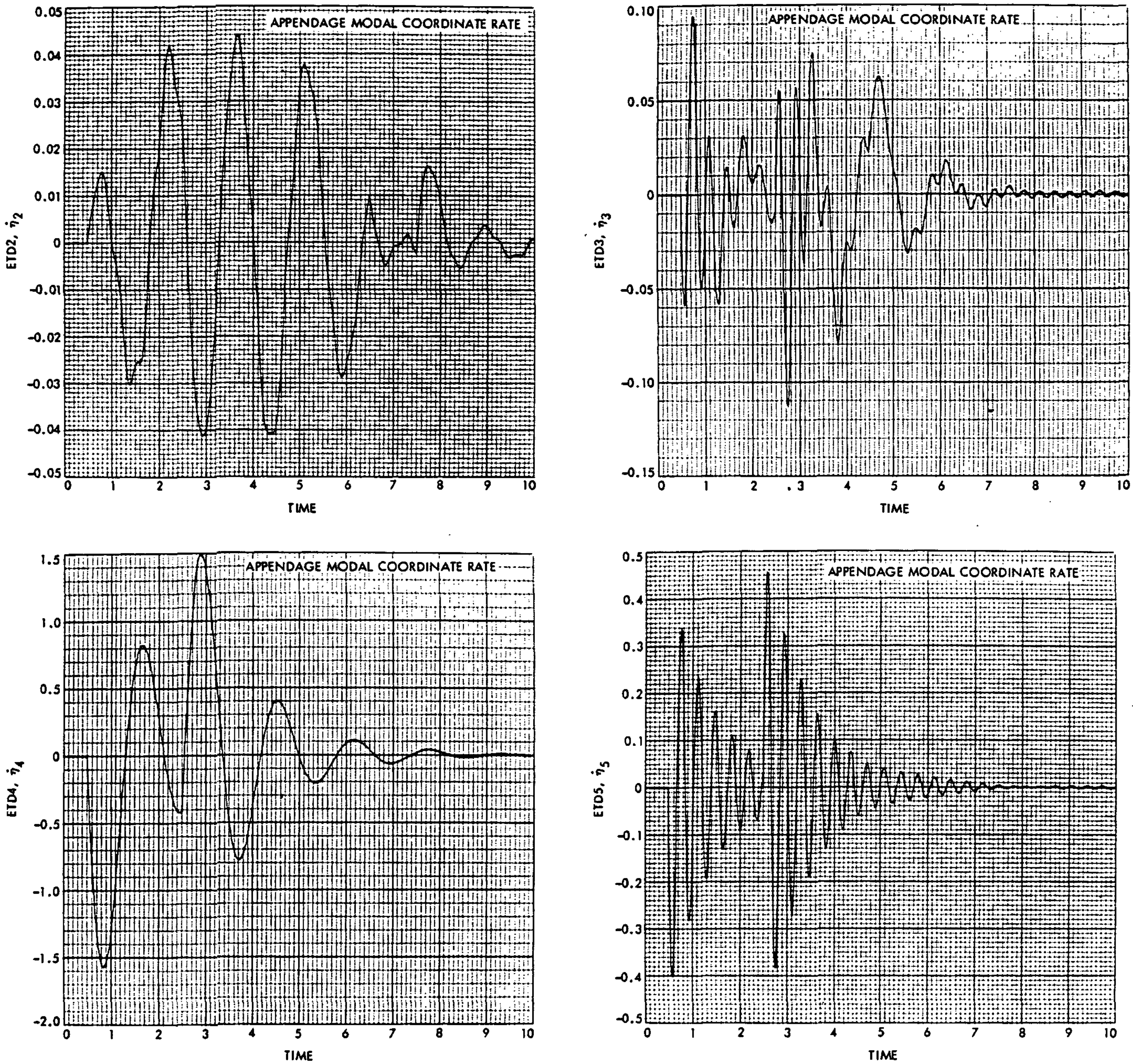
APPENDAGE MODAL COORDINATE RATE
ETD2, $\dot{\eta}_2$
TIME
APPENDAGE MODAL COORDINATE RATE
ETD3, $\dot{\eta}_3$
TIME
APPENDAGE MODAL COORDINATE RATE
ETD4, $\dot{\eta}_4$
TIME
APPENDAGE MODAL COORDINATE RATE
ETD5, $\dot{\eta}_5$
TIME

Fig. 12 (contd)

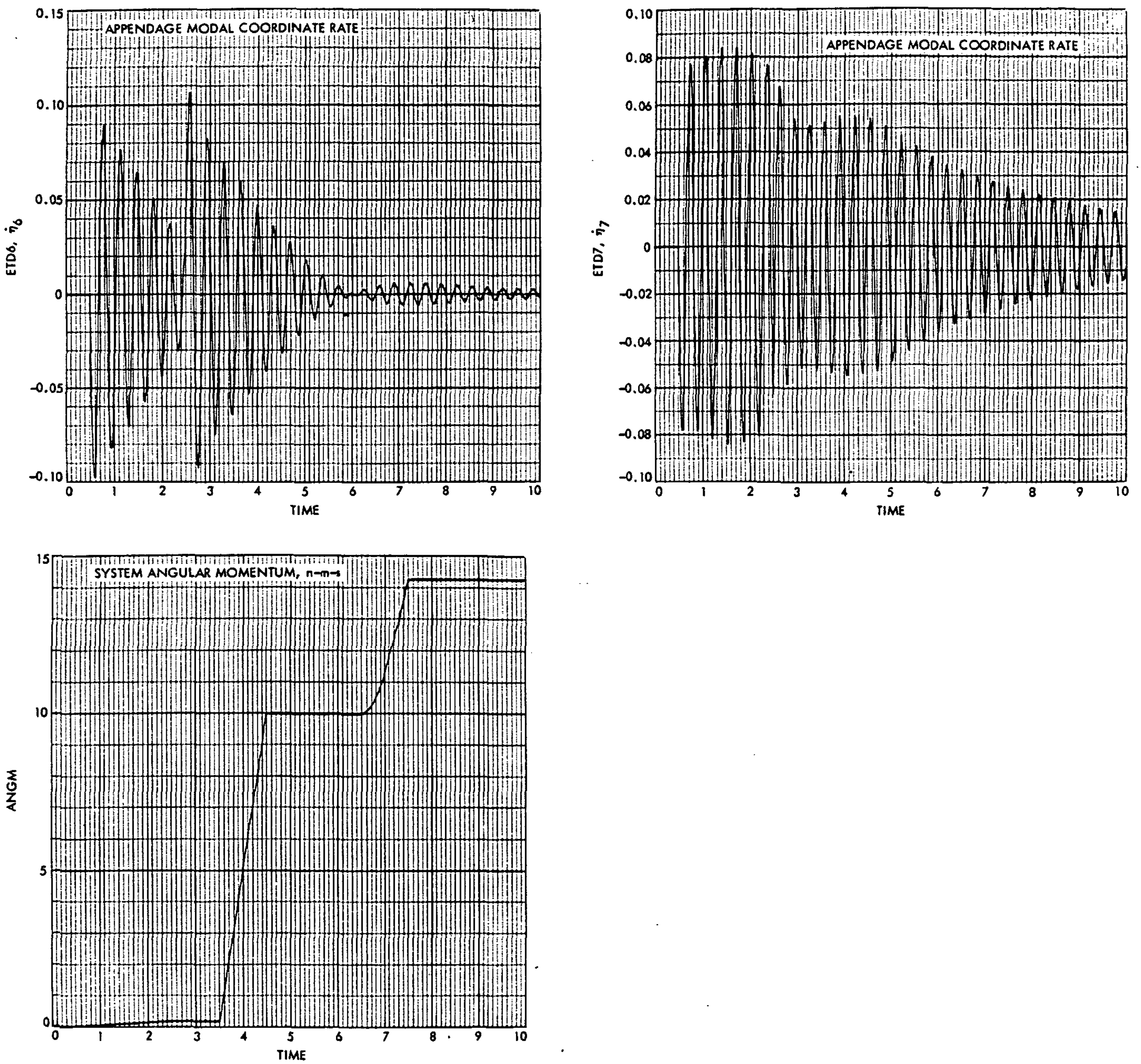
APPENDAGE MODAL COORDINATE RATE
ETD6, $\dot{\eta}_6$
TIME
APPENDAGE MODAL COORDINATE RATE
ETD7, $\dot{\eta}_7$
TIME
SYSTEM ANGULAR MOMENTUM, n-m-s
ANGM
TIME

Fig. 12 (contd)

Exactly the same simulation can be made using the linearized subroutine version, MBDYFL. The only change necessary in the simulation program of Fig. 11 to allow the use of the linearized version is the change of "CALL MBDYFN(NC, . . .)" to "CALL MBDYFL(NC, . . .)" in the initialization section. This was done and resulted in solutions for the system response which are virtually indistinguishable from those plotted in Fig. 12. However, some slight deviations are detectable in the printed output shown in Fig. 14 when compared with the MBDYFN results of Fig. 13. The major difference between the two simulations in this case is reflected in the computer running time. A total of 2 min of accountable central processor time (Univac 1108) was required by the program using MBDFN as contrasted with only 1 min of central processor time used by the MBDYFL program. In addition, memory storage is considerably reduced by the use of MBDYFL, so that the overall cost of producing the desired solutions in this case is significantly reduced.

Another convenient method of reducing computation time and therefore cost under certain circumstances is to use these subroutines' prescribed variable option. By setting PI(i) = 1, the hinge angle variables γ_i, $\dot{\gamma}_i$, and $\ddot{\gamma}_i$ may be prescribed, i.e., defined by the user in the main program rather than computed within the subroutine. When this is done, any expression in the main program defining the hinge torque τ_i(TH(i)) is ignored by the subroutine. The equations normally solved by the subroutine to obtain $\ddot{\gamma}_i$ are then deleted from consideration, thus reducing the system order and speeding up calculations.

For an example of this approach, we can return to the program of Fig. 11, using MBDYFN, and change PI so that PI(1) = 1 and PI(2) = 1 (leaving PI(3) = 1 unchanged so that the angular momentum calculation is still performed), as shown in Fig. 15. This means that the platform hinge rotations are to be prescribed. However, by not defining any function for GMDD(1) and GMDD(2), these variables remain zero, as will their integrals. Thus, the simulation will proceed as before but with $\ddot{\gamma}_i = \dot{\gamma}_i = \gamma_i = 0$ (i = 1, 2); i.e., the platform will be "frozen" or rigidly connected to $\mathscr{B}_0$.

The system response (with identical disturbances) in this configuration was simulated, and the plotted results were indistinguishable from those in Fig. 13. A sample of the simulation's printed output, shown in Fig. 16, indicates clearly that "freezing" the platform has had no significant effect on the dynamic response of the reference body or the appendage modal coordinates. However, some numerical differences are discernible in the printout.

Thus, prescribing the platform's "motion" in this case did not appreciably change the overall result and, as a matter of fact, took 15 s less computation time than the original run with no prescribed variables, a saving of $\frac{1}{8}$.

V. Summary and Conclusions

In this report, detailed mathematical models have been developed, suitable for describing the attitude dynamics of vehicles that may be idealized as systems of interconnected rigid bodies with possible terminal flexible appendages. The resulting mathematical formulations apply to two kinds of system behavior: (1) generally arbitrary rigid-body rotations with the restriction that appendage base body deviations from some nominal constant spin rate are small, and (2) unrestrained rigid-body rotations with the restriction that appendage base motion deviations

```
TIME  =  2.20000    W1    = -5.534412-04      W2    =  4.144790-04      W3    = -6.254046-05
                    NX    =  0.000000         NY    =  0.000000         FZ    =  300.000
                    ETA1  =  .175147          ETA2  = -1.585589-03      ETA3  =  3.189027-03
                    ETA4  = -.388847          ETA5  = -2.479597-02      ETA6  = -5.228946-03
                    ETA7  = -4.007270-03      ETD1  = -1.014957-02      ETD2  =  4.176284-02
                    ETD3  =  1.388555-02      ETD4  = -.153645          ETD5  =  7.823737-02
                    ETD6  =  3.289947-02      ETD7  = -7.956042-02      ANGM  =  .147617
                    W1D   =  5.287033-04      W2D   = -5.007478-03      W3D   = -2.201360-04
                    GM1   =  1.434090-06      GM1D  = -5.751964-06      GM2   =  1.725330-05
                    GM2D  =  5.170688-05

TIME  =  2.30000    W1    = -1.194999-04      W2    =  2.035237-04      W3    =  9.543952-06
                    NX    =  0.000000         NY    =  0.000000         FZ    =  300.000
                    ETA1  =  .178812          ETA2  =  2.452690-03      ETA3  =  3.883458-03
                    ETA4  = -.412792          ETA5  = -2.078913-02      ETA6  = -4.532183-03
                    ETA7  = -7.834672-03      ETD1  =  7.930726-02      ETD2  =  3.725216-02
                    ETD3  = -1.929868-03      ETD4  = -.315038          ETD5  = -1.657669-02
                    ETD6  = -2.085271-02      ETD7  =  2.710963-02      ANGM  =  .154926
                    W1D   =  6.476792-03      W2D   = -1.162214-03      W3D   =  1.115572-03
                    GM1   = -7.351207-08      GM1D  = -1.607631-05      GM2   =  2.053021-06
                    GM2D  = -2.875026-04

TIME  =  2.40000    W1    =  3.141630-04      W2    = -1.896092-04      W3    =  4.590424-05
                    NX    =  0.000000         NY    =  0.000000         FZ    =  300.000
                    ETA1  =  .190410          ETA2  =  5.807521-03      ETA3  =  2.901173-03
                    ETA4  = -.449504          ETA5  = -2.620123-02      ETA6  = -7.185827-03
                    ETA7  = -1.473263-03      ETD1  =  .150234          ETD2  =  3.053111-02
                    ETD3  = -1.494873-02      ETD4  = -.406566          ETD5  = -6.151531-02
                    ETD6  = -1.660914-02      ETD7  =  5.842808-02      ANGM  =  .162645
                    W1D   =  5.194020-04      W2D   = -7.042102-03      W3D   = -6.531757-04
                    GM1   = -4.531444-07      GM1D  =  9.725387-06      GM2   = -1.994955-05
                    GM2D  = -6.297991-05

TIME  =  2.50000    W1    = -2.551416-05      W2    = -8.965070-04      W3    = -7.507707-05
                    NX    =  0.000000         NY    =  0.000000         FZ    =  300.000
                    ETA1  =  .208125          ETA2  =  8.598550-03      ETA3  =  1.576923-03
                    ETA4  = -.491649          ETA5  = -2.791420-02      ETA6  = -6.540928-03
                    ETA7  = -1.978658-03      ETD1  =  .198350          ETD2  =  2.431256-02
                    ETD3  = -9.086693-03      ETD4  = -.425067          ETD5  =  3.388802-02
                    ETD6  =  2.364968-02      ETD7  = -6.403697-02      ANGM  =  .171715
                    W1D   = -5.547869-03      W2D   = -4.766604-03      W3D   = -1.194857-03
                    GM1   =  9.270155-07      GM1D  =  1.021694-05      GM2   = -6.263739-06
                    GM2D  =  2.760264-04
```

Fig. 13. Simulation printout for program using MBDYFN

```
TIME  =  2.20000     W1    = -5.534418-04     W2    =  4.144801-04     W3    = -6.254079-05
                     NX    =  0.000000        NY    =  0.000000        FZ    =  300.000
                     ETA1  =   .175147        ETA2  = -1.585590-03     ETA3  =  3.189027-03
                     ETA4  =  -.388847        ETA5  = -2.479597-02     ETA6  = -5.228946-03
                     ETA7  = -4.007270-03     ETD1  = -1.014957-02     ETD2  =  4.175283-02
                     ETD3  =  1.388555-02     ETD4  =  -.153645        ETD5  =  7.823734-02
                     ETD6  =  3.289947-02     ETD7  = -7.956043-02     ANGM  =   .147618
                     W1D   =  5.287010-04     W2D   = -5.007477-03     W3D   = -2.201367-04
                     GM1   =  1.433806-06     GM1D  = -5.756394-06     GM2   =  1.725269-05
                     GM2D  =  5.171269-05

TIME  =  2.30000     W1    = -1.195008-04     W2    =  2.035250-04     W3    =  9.543577-06
                     NX    =  0.000000        NY    =  0.000000        FZ    =  300.000
                     ETA1  =   .178812        ETA2  =  2.452688-03     ETA3  =  3.883458-03
                     ETA4  =  -.412792        ETA5  = -2.078914-02     ETA6  = -4.532183-03
                     ETA7  = -7.834672-03     ETD1  =  7.930725-02     ETD2  =  3.725216-02
                     ETD3  = -1.929866-03     ETD4  =  -.315038        ETD5  = -1.657669-02
                     ETD6  = -2.085271-02     ETD7  =  2.710963-02     ANGM  =   .154927
                     W1D   =  6.476788-03     W2D   = -1.142213-03     W3D   =  1.115572-03
                     GM1   = -7.378356-08     GM1D  = -1.607390-05     GM2   =  2.052887-06
                     GM2D  = -2.874996-04

TIME  =  2.40000     W1    =  3.141622-04     W2    = -1.896078-04     W3    =  4.590388-05
                     NX    =  0.000000        NY    =  0.000000        FZ    =  300.000
                     ETA1  =   .190410        ETA2  =  5.807519-03     ETA3  =  2.901173-03
                     ETA4  =  -.449504        ETA5  = -2.620123-02     ETA6  = -7.185827-03
                     ETA7  = -1.473263-03     ETD1  =   .150234        ETD2  =  3.053111-02
                     ETD3  = -1.494873-02     ETD4  =  -.406566        ETD5  = -6.151530-02
                     ETD6  = -1.660914-02     ETD7  =  5.842808-02     ANGM  =   .162647
                     W1D   =  5.194041-04     W2D   = -7.042101-03     W3D   = -6.531752-04
                     GM1   = -4.530906-07     GM1D  =  9.730052-06     GM2   = -1.994953-05
                     GM2D  = -6.297931-05

TIME  =  2.50000     W1    = -2.551484-05     W2    = -8.965056-04     W3    = -7.507740-05
                     NX    =  0.000000        NY    =  0.000000        FZ    =  300.000
                     ETA1  =   .208125        ETA2  =  8.598547-03     ETA3  =  1.576924-03
                     ETA4  =  -.491649        ETA5  = -2.791420-02     ETA6  = -6.540928-03
                     ETA7  = -1.978658-03     ETD1  =   .198350        ETD2  =  2.431256-02
                     ETD3  = -9.086695-03     ETD4  =  -.425067        ETD5  =  3.388802-02
                     ETD6  =  2.364968-02     ETD7  = -6.403697-02     ANGM  =   .171716
                     W1D   = -5.547869-03     W2D   = -4.766604-03     W3D   = -1.194857-03
                     GM1   =  9.273785-07     GM1D  =  1.021708-05     GM2   = -6.263727-06
                     GM2D  =  2.760250-04
```

Fig. 14. Simulation printout for program using MBDYFL

```
CSSL III  JET PROPULSION LABORATORY  040374-A002H  021775-225735

*** START                T(RUN)=  17.986 T(TASK)=     .003  CTP   =     .535
                                        DT(TASK)=     .003  DCTP  =     .535

    PROGRAM   2-BODY VEHICLE WITH FLEX. APPENDAGE (FROZEN PLATFORM)
    *SC4020   BLDG/198,BOX/601,CAMERA/9IN,FRAMES/50
    COMMENT
          ARRAY MB(7),MS(1,7),PB(3),PS(1,1,3),G(2,3)
          ARRAY EIG(1,6,7),RF(1,1,3),REC(1,6,7),WF(1,7),ZF(1,7)
          ARRAY TB(3),TS(1,3),FB(3),FS(1,3),GM(2),GMD(2),GMDD(2)
          ARRAY TH(2),WO(3),TF(1,1,3),FF(1,1,3),ET(1,7),ETD(1,7)
          DOUBLE PRECISION WDOT(5),ETDD(1,7)
          INTEGER NC,NF,H(1,2),F(1,3),PI(3),L
    DATA H(1,1)/0/H(1,2)/2/PI/1,1,1/
    DATA F(1,1)/0/F(1,2)/1/F(1,3)/7/
    DATA MB/1230.,1290.,1650.,-16.29,-43.45,61.75,79.0/
    DATA MS/4.75,5.53,1.32,0.,0.,0.,1.93/
    DATA G(1,3)/1./G(2,1)/1./
    DATA REC/.03375,.001654,-.8678,.08135,17.17,.007955,...
             .01065,.001104,-.4608E-4,.4236,12.3,.001859,...
             .002335,-.01818,-.4731E-5,21.3,-.2386,.000918,...
             .003244,.001014,2.234,-.4081,5.930,-.05211,...
             -.6055,-.5381,1.962,7.577,.4020,2.520,...
             -.3060,1.753,.5585,-4.32,-.1589,-.9205,...
             -.02762,.1051,.3919,2.032,-2.061,.2761/
    DATA WF/.5756,.61337,.61337,.63071,2.723,2.963,3.047/
    DATA ZF/.20,.20,.20,.20,.05,.05,.01/
          CONSTANT FINTIM=10.,CLKTIM=900.,PIE=3.14159265
          CONSTANT K1=900.,B1=100.,K2=850.,B2=100.
    INITIAL
          NC=1   $   NF=1
          DO 57 L=1,7
    57..  WF(1,L)=WF(1,L)*2.*PIE
          CALL MBDYFN(NC,H,MB,MS,PB,PS,G,PI,NF,F,EIG,REC,RF,WF,ZF)
    END
    DYNAMIC
          IF(TIME.GT.FINTIM) GO TO FIN
          STPCLK   CLKTIM
          OUTPUT 10,W1,W2,W3,NX,NY,FZ,ETA1,ETA2,ETA3,ETA4,ETA5,ETA6,ETA7,...
                     ETD1,ETD2,ETD3,ETD4,ETD5,ETD6,ETD7,ANGM,W1D,W2D,W3D,...
                     GM1,GM1D,GM2,GM2D
          PREPAR W1,W2,W3,NX,NY,FZ,ETD1,ETD2,ETD3,ETD4,ETD5,ETD6,ETD7,...
                     ANGM,GM1,GM2,GM1D,GM2D
    DERIVATIVE   BODY2F
          VARIABLE TIME=0.  $  CINTERVAL  CI=.01
          XERROR W1=1.E-6     $     MERROR W1=1.E-6
    NOSORT
          GMD(1)=GM1D   $   GM(1)=GM1
          GMD(2)=GM2D   $   GM(2)=GM2
          ET(1,1)=ETA1 $ ET(1,2)=ETA2 $ ET(1,3)=ETA3 $ ET(1,4)=ETA4
          ET(1,5)=ETA5 $ ET(1,6)=ETA6 $ ET(1,7)=ETA7
          ETD(1,1)=ETD1 $ ETD(1,2)=ETD2 $ ETD(1,3)=ETD3 $ ETD(1,4)=ETD4
          ETD(1,5)=ETD5 $ ETD(1,6)=ETD6 $ ETD(1,7)=ETD7
          WO(1)=W1  $  WO(2)=W2  $  WO(3)=W3  $  ANGM=HM
    COMMENT...
                  HINGE TORQUES
```

Fig. 15. Simulation program for test vehicle with prescribed platform motion using MBDYFN

```
COMMENT
       TH(1)=-K1*GM1 - B1*GM1D
       TH(2)=-K2*GM2 - B2*GM2D
COMMENT...
                  FORCE EQUATION
COMMENT
       FZ=(STEP(.5,TIME)-STEP(2.5,TIME))*300.
       FB(3)=FZ
COMMENT...
                  ENGINE TORQUE
COMMENT
       NX=(STEP(3.5,TIME)-STEP(4.5,TIME))*10.
       NY=(STEP(6.5,TIME)-STEP(7.5,TIME))*10.
       TB(1)=NX   $   TB(2)=NY
COMMENT...
                  SOLUTION FOR SYSTEM ACCELERATIONS
COMMENT
       CALL MRATE(NC,TH,TB,TS,FB,FS,TF,FF,GM,GMD,GMDD,ET,ETD,W0,WDOT,...
       ETDD,HM)
       W1D=WDOT(1)   $   W2D=WDOT(2)   $   W3D=WDOT(3)
COMMENT...
                  SYSTEM RATES AND POSITIONS
COMMENT
       W1=INTEG(WDOT(1),0.)
       W2=INTEG(WDOT(2),0.)
       W3=INTEG(WDOT(3),0.)
       ETD1=INTEG(ETDD(1,1),0.)    $    ETA1=INTEG(ETD1,0.)
       ETD2=INTEG(ETDD(1,2),0.)    $    ETA2=INTEG(ETD2,0.)
       ETD3=INTEG(ETDD(1,3),0.)    $    ETA3=INTEG(ETD3,0.)
       ETD4=INTEG(ETDD(1,4),0.)    $    ETA4=INTEG(ETD4,0.)
       ETD5=INTEG(ETDD(1,5),0.)    $    ETA5=INTEG(ETD5,0.)
       ETD6=INTEG(ETDD(1,6),0.)    $    ETA6=INTEG(ETD6,0.)
       ETD7=INTEG(ETDD(1,7),0.)    $    ETA7=INTEG(ETD7,0.)
       GM1D=INTEG(WDOT(4),0.)    $    GM1=INTEG(GM1D,0.)
       GM2D=INTEG(WDOT(5),0.)    $    GM2=INTEG(GM2D,0.)
END
END
END
TERMINAL
FIN.. CONTINUE
END
END
```

Fig. 15 (contd)

ORIGINAL PAGE IS
OF POOR QUALITY

```
TIME  =  2.20000   W1    = -5.544373-04    W2    =  4.139068-04    W3    = -6.283681-05
                   NX    =  0.000000       NY    =  0.000000       FZ    =  300.000
                   ETA1  =  .175147        ETA2  = -1.585691-03    ETA3  =  3.192810-03
                   ETA4  = -.388847        ETA5  = -2.478928-02    ETA6  = -5.240584-03
                   ETA7  = -3.998999-03    ETD1  = -1.013886-02    ETD2  =  4.176088-02
                   ETD3  =  1.390250-02    ETD4  = -.153641        ETD5  =  7.847859-02
                   ETD6  =  3.297854-02    ETD7  = -7.973409-02    ANGM  =  .147619
                   W1D   =  5.576106-04    W2D   = -5.012866-03    W3D   = -2.146401-04
                   GM1   =  0.000000       GM1D  =  0.000000       GM2   =  0.000000
                   GM2D  =  0.000000

TIME  =  2.30000   W1    = -1.184182-04    W2    =  2.034345-04    W3    =  9.839955-06
                   NX    =  0.000000       NY    =  0.000000       FZ    =  300.000
                   ETA1  =  .178813        ETA2  =  2.453164-03    ETA3  =  3.885903-03
                   ETA4  = -.412792        ETA5  = -2.077761-02    ETA6  = -4.525558-03
                   ETA7  = -7.845880-03    ETD1  =  7.930792-02    ETD2  =  3.725211-02
                   ETD3  = -1.960251-03    ETD4  = -.315037        ETD5  = -1.672914-02
                   ETD6  = -2.068424-02    ETD7  =  2.702194-02    ANGM  =  .154925
                   W1D   =  6.476257-03    W2D   = -1.150093-03    W3D   =  1.118775-03
                   GM1   =  0.000000       GM1D  =  0.000000       GM2   =  0.000000
                   GM2D  =  0.000000

TIME  =  2.40000   W1    =  3.138598-04    W2    = -1.889694-04    W3    =  4.603753-05
                   NX    =  0.000000       NY    =  0.000000       FZ    =  300.000
                   ETA1  =  .190410        ETA2  =  5.807416-03    ETA3  =  2.901913-03
                   ETA4  = -.449504        ETA5  = -2.620992-02    ETA6  = -7.179536-03
                   ETA7  = -1.473728-03    ETD1  =  .150223        ETD2  =  3.052319-02
                   ETD3  = -1.494876-02    ETD4  = -.406570        ETD5  = -6.166167-02
                   ETD6  = -1.677170-02    ETD7  =  5.865821-02    ANGM  =  .162643
                   W1D   =  5.074690-04    W2D   = -7.044410-03    W3D   = -6.582883-04
                   GM1   =  0.000000       GM1D  =  0.000000       GM2   =  0.000000
                   GM2D  =  0.000000

TIME  =  2.50000   W1    = -2.521466-05    W2    = -8.968403-04    W3    = -7.534413-05
                   NX    =  0.000000       NY    =  0.000000       FZ    =  300.000
                   ETA1  =  .208125        ETA2  =  8.598212-03    ETA3  =  1.577519-03
                   ETA4  = -.491649        ETA5  = -2.792129-02    ETA6  = -6.549905-03
                   ETA7  = -1.967483-03    ETD1  =  .198357        ETD2  =  2.431720-02
                   ETD3  = -9.100753-03    ETD4  = -.425064        ETD5  =  3.404423-02
                   ETD6  =  2.359918-02    ETD7  = -6.410951-02    ANGM  =  .171715
                   W1D   = -5.530066-03    W2D   = -4.777549-03    W3D   = -1.195949-03
                   GM1   =  0.000000       GM1D  =  0.000000       GM2   =  0.000000
                   GM2D  =  0.000000
```

Fig. 16. Simulation printout for program using MBDYFN with prescribed platform motion

from a nominally *zero* angular rate are small. The second approach was then further restricted to the often very useful assumption that *all* system rotations are small, permitting a formal linearization with respect to hinge and reference body rotations. Of course, appendage deformations are assumed small in every case.

Three FORTRAN subroutines were then described which solve the equations of motion for these three cases, namely, MBDYFR (for spinning appendages), MBDYFN (for nonspinning appendages), and MBDYFL (linearized for small rotations). Each of the routines has much the same functional appearance as those programs described in Ref. 6., i.e., an initializing entry and a dynamic entry point, with the only differences being the addition of *appendage-related* parameters, variables, and forcing functions. The routines also retain the option of user-prescribed rotations at selected hinge connections. However, an additional option provided in these programs is that of calculating angular momentum magnitude, which at times provides a valuable check on computational accuracy.

In applying MBDYFR, one can conclude that the mathematical difficulties introduced by spin have forced not only a first-order transformation to obtain uncoupled coordinates but, as a consequence, *two* coordinates per mode must be solved for in the subroutine. However, what appears to be a computational disadvantage in this case may well be softened by the necessity to consider fewer modes. Some other difficulties are also introduced by this particular modal transformation. The presence of *both* the modal coordinate position and rate in the expressions for appendage deformation and deformation rate can lead to significant error if modal damping is inserted (thus disturbing eigenvector orthogonality) *and* large steady-state appendage deformations are present. The user must ensure that any appendage deformations in the damped case remain essentially oscillatory about a nominally zero mean. MBDYFR, as it now stands, also forces the user, regardless of which appendages are spinning or not spinning, to formulate each appendage's modal description using only the first-order transformation, i.e., as if it were subject to spin. While it was much more convenient to program MBDYFR in this way, future requirements for improved computational efficiency may make a modification of MBDYFR desirable. Still, in spite of these particular characteristics, it is felt that MBDYFR can be successfully employed in a wide variety of applications because of its inherent generality and versatility. In addition to the prescribed variable and angular momentum calculation options, the user may also choose to use MBDYFR to directly calculate the steady-state deformations due to centrifugal forces. This is accomplished by setting SR, the nominal appendage spin rate, to zero even though, in the simulation, the appendage is spinning. Setting SR to zero *restores* the centrifugal force terms to the equations, and appropriate deformations will appear in the solution. However, as indicated before, the greater the modal damping under these circumstances, the larger the numerical error will be in the steady-state deformations due to spin.

The routines MBDYFN and MBDYFL are of more immediate utility at JPL since current spacecraft designs here are three-axis-stabilized. They represent a generalization of the hybrid-mode concept, developed in Ref. 2, to the rigid-body-tree approach. As a result, it is no longer necessary to add special terms and re-derive equations of motion in order to accommodate discrete rigid-body rotations (or translations) in the system (as was done, for example, in Ref. 9 for the Viking Orbiter with flexible appendages and rigid propellant slosh masses). Even translational dampers can be reasonably well approximated within the hinge-connected tree system. Because of its speed advantages and because it usually provides acceptable solution accuracy even when rotations are not strictly small, the completely linearized version, MBDYFL, will offer the greatest utility among the three programs at JPL for routine control design studies.

To make these subroutines more easily available to the aerospace industry, they have been submitted to COSMIC (Computer Software Management and Information Center), University of Georgia, Athens, Georgia, for evaluation and dissemination to interested agencies and institutions.

References

1. Likins, P.W., "Finite Element Appendage Equations for Hybrid Coordinate Dynamic Analysis," *International Journal of Solids and Structures*, Vol. 8, pp. 709–731, 1972. (Also, in expanded form, Technical Report 32-1525, Jet Propulsion Laboratory, Pasadena, Calif., Oct. 15, 1971.)
2. Likins, P.W., *Dynamics and Control of Flexible Space Vehicles*, Technical Report 32-1329, Jet Propulsion Laboratory, Pasadena, Calif., Jan. 15, 1970.
3. Likins, P.W., and Ohkami, Y., "Coordinate Transformation and Truncation for Rotating Spacecraft With Flexible Appendages," AIAA Journal (to be published).
4. Likins, P.W., "Dynamic Analysis of a System of Hinge-Connected Rigid Bodies With Non-Rigid Appendages," *Int. J. Solids and Structures*, Vol. 9, pp. 1473–1488, 1973. (Also, in expanded form, Technical Report 32-1576, Jet Propulsion Laboratory, Pasadena, Calif., Feb. 1, 1974.)
5. Hooker, W.W., *Equations of Attitude Motion of a Topological Tree of Bodies, the Terminal Members of Which May Be Flexible*, Report LMSC-D354938, Lockheed Palo Alto Research Laboratory, Palo Alto, Calif., Nov. 1, 1973.
6. Fleischer, G.E., and Likins, P.W., *Attitude Dynamics Simulation Subroutines for Systems of Hinge-Connected Rigid Bodies*, Technical Report 32-1592, Jet Propulsion Laboratory, Pasadena, Calif., May 1, 1974.
7. Fleischer, G.E., *Attitude Dynamics Simulation Subroutine Modifications*, Technical Memorandum, Jet Propulsion Laboratory, Pasadena, Calif. (to be published).
8. Likins, P.W., and Fleischer, G.E., *Large-Deformation Modal Coordinates for Nonrigid Vehicle Dynamics*, Technical Report 32-1565, Jet Propulsion Laboratory, Pasadena, Calif., Nov. 1, 1972.
9. Fleischer, G.E., and McGlinchey, L.F., "Viking Thrust Vector Control Dynamics Using Hybrid Coordinates to Model Vehicle Flexibility and Propellant Slosh," AAS paper No. 71-348, presented at AAS/AIAA Astrodynamics Specialists Conference, Ft. Lauderdale, Fla., Aug. 17–19, 1971.

Appendix A

Effects of Damping on Rotating Appendage Equations

In Section IIIA, it was pointed out that the addition of viscous damping-like terms to the already transformed appendage equations, particularly for the case of a nominally rotating appendage/base, is *mathematically* not justified. However, the insertion of modal damping terms is usually thought to be justified on the practical basis that it reasonably and more conveniently represents the *physical* response of systems as determined from actual test data.

However, it may be useful to illustrate how and to what extent the mathematical inconsistencies so introduced may affect computational results. For example, one can show that the insertion of modal damping into Eq. (26) introduces errors in the *steady-state* values of $\bar{\delta}^k$, $\bar{\eta}^k$, and therefore the deformations q^k *and* $\dot{q}^k$. This can be seen from the following. Repeating Eqs. (26) and (27), we have

$$\dot{\bar{\delta}}^k = -\bar{\sigma}^k\bar{\eta}^k - \bar{\sigma}^k\bar{\Gamma}_k^T L_k' - \bar{\xi}^k\bar{\sigma}^k\bar{\delta}^k \qquad \text{(A-1)}$$

$$\dot{\bar{\eta}}^k = \bar{\sigma}^k\bar{\delta}^k - \bar{\sigma}^k\bar{\psi}_k^T L_k' - \bar{\xi}^k\bar{\sigma}^k\bar{\eta}^k \qquad \text{(A-2)}$$

$$q^k = 2\left(\bar{\psi}_k\bar{\delta}^k - \bar{\Gamma}_k\bar{\eta}^k\right) \qquad \text{(A-3)}$$

$$\dot{q}^k = -2\left(\bar{\Gamma}_k\bar{\sigma}^k\bar{\delta}^k + \bar{\psi}_k\bar{\sigma}^k\bar{\eta}^k\right) \qquad \text{(A-4)}$$

$$\ddot{q}^k = -2\left(\bar{\Gamma}_k\bar{\sigma}^k\dot{\bar{\delta}}^k + \bar{\psi}_k\bar{\sigma}^k\dot{\bar{\eta}}^k\right) \qquad \text{(A-5)}$$

If we now examine q^k and $\dot{q}^k$ when $\bar{\delta}^k$ and $\bar{\eta}^k$ have reached a steady-state condition, i.e., when $\dot{\bar{\delta}}^k = \dot{\bar{\eta}}^k = 0$, we have, from (A-1),

$$\bar{\sigma}^k\bar{\eta}^k = -\bar{\sigma}^k\bar{\Gamma}_k^T L_k' - \bar{\xi}^k\bar{\sigma}^k\bar{\delta}^k$$

and from (A-2),

$$\bar{\sigma}^k\bar{\delta}^k = \bar{\sigma}^k\bar{\psi}_k^T L_k' + \bar{\xi}^k\bar{\sigma}^k\bar{\eta}^k$$

Substituting from (A-2) into (A-1),

$$\bar{\eta}^k = -\bar{\Gamma}_k^T L_k' - \bar{\sigma}^{k^{-1}}\bar{\xi}^k\bar{\sigma}^k\left[\bar{\psi}_k^T L_k' + \bar{\sigma}^{k^{-1}}\bar{\xi}^k\bar{\sigma}^k\bar{\eta}^k\right]$$

or

$$\bar{\eta}^k = -\bar{\Gamma}_k^T L_k' - \bar{\xi}^k\bar{\psi}_k^T L_k' - \bar{\xi}^k\bar{\xi}^k\bar{\eta}^k$$

or

$$\bar{\eta}_{ss}^{k} = \left(U + \bar{\xi}^{k}\bar{\xi}^{k}\right)^{-1}\left(-\bar{\Gamma}_{k}^{T} - \bar{\xi}^{k}\bar{\psi}_{k}^{T}\right)L'_{k_{ss}} \tag{A-6}$$

Substituting from (A-1) into (A-2),

$$\bar{\delta}^{k} = \bar{\psi}_{k}^{T}L'_{k} + \bar{\sigma}^{k^{-1}}\bar{\xi}^{k}\bar{\sigma}^{k}\left[-\bar{\Gamma}_{k}^{T}L'_{k} - \bar{\sigma}^{k^{-1}}\bar{\xi}^{k}\bar{\sigma}^{k}\bar{\delta}^{k}\right]$$

or

$$\bar{\delta}^{k} = \bar{\psi}_{k}^{T}L'_{k} - \bar{\xi}^{k}\bar{\Gamma}_{k}^{T}L'_{k} - \bar{\xi}^{k}\bar{\xi}^{k}\bar{\delta}^{k}$$

or

$$\bar{\delta}_{ss}^{k} = \left(U + \bar{\xi}^{k}\bar{\xi}^{k}\right)^{-1}\left(\bar{\psi}_{k}^{T} - \bar{\xi}^{k}\bar{\Gamma}_{k}^{T}\right)L'_{k_{ss}} \tag{A-7}$$

From (A-3), (A-6), and (A-7),

$$q_{ss}^{k} = 2\bar{\psi}_{k}\left(U + \bar{\xi}^{k}\bar{\xi}^{k}\right)^{-1}\left(\bar{\psi}_{k}^{T} - \bar{\xi}^{k}\bar{\Gamma}_{k}^{T}\right)L'_{k} - 2\bar{\Gamma}_{k}\left(U + \bar{\xi}^{k}\bar{\xi}^{k}\right)^{-1}\left(-\bar{\Gamma}_{k}^{T} - \bar{\xi}^{k}\bar{\psi}_{k}^{T}\right)L'_{k}$$

$$q_{ss}^{k} = 2\left[\bar{\psi}_{k}U_{\xi}^{-1}\left(\bar{\psi}_{k}^{T} - \bar{\xi}^{k}\bar{\Gamma}_{k}^{T}\right) + \bar{\Gamma}_{k}U_{\xi}^{-1}\left(\bar{\Gamma}_{k}^{T} + \bar{\xi}^{k}\bar{\psi}_{k}^{T}\right)\right]L'_{k_{ss}} \tag{A-8}$$

where

$$U_{\xi} = \left(U + \bar{\xi}^{k}\bar{\xi}^{k}\right)$$

From (A-4), (A-6), and (A-7),

$$\dot{q}_{ss}^{k} = 2\left[\bar{\psi}_{k}\bar{\sigma}^{k}U_{\xi}^{-1}\left(\bar{\Gamma}_{k}^{T} + \bar{\xi}^{k}\bar{\psi}_{k}^{T}\right) - \bar{\Gamma}_{k}\bar{\sigma}^{k}U_{\xi}^{-1}\left(\bar{\psi}_{k}^{T} - \bar{\xi}^{k}\bar{\Gamma}_{k}^{T}\right)\right]L'_{k_{ss}} \tag{A-9}$$

Notice that from (A-9), $\dot{q}_{ss}^{k} \neq 0$ in general! However, as ξ^{k} becomes infinitesimally small, (A-8) and (A-9) approach

$$q_{ss}^{k} = 2\left[\bar{\psi}_{k}\bar{\psi}_{k}^{T} + \bar{\Gamma}_{k}\bar{\Gamma}_{k}^{T}\right]L'_{k}$$

and

$$\dot{q}_{ss}^{k} = 2\left[\bar{\psi}_{k}\bar{\sigma}^{k}\bar{\Gamma}_{k}^{T} - \bar{\Gamma}_{k}\bar{\sigma}^{k}\bar{\psi}_{k}^{T}\right]L'_{k} \equiv 0$$

due to orthogonality relations between $\bar{\psi}_{k}$ and $\bar{\Gamma}_{k}$.

The discovery above that, in general, $\dot{q}_{ss}^{k} \neq 0$ when modal damping is introduced is rather disconcerting. It is further disturbing to realize that if the appendage deformation rates $\dot{q}^{k}$ are not zero when the *modal* coordinates appear to indicate an appendage *at rest*, then the angular momentum calculations of the subroutines, based on $\dot{q}^{k}$, will be in error as well.

Fortunately, we have assumed that the appendage deformations, q^k, and their derivatives are small *and* represent only the oscillatory component of the total possible deformation. This tends to imply that L_k' must be very small to begin with and that the steady-state levels of q^k (or its derivatives) after damping are "small" compared to its *transient* oscillatory amplitudes. Therefore the errors introduced in (A-8) and (A-9) should be of relatively little significance. However, one should be aware of their existence and that they can add to other computational errors.

Appendix B

System Angular Momentum Computation

In Ref. 5, Hooker shows that for a dynamical system of the type considered here, namely, a topological tree of rigid bodies any one of which may carry a flexible appendage, the equations are of the general form

$$A\dot{x} = B$$

where

$$A = \left[\begin{array}{c|c|c} a_{00} & a_{0k} & b_0 \\ \hline a_{0k}^T & a & b \\ \hline b_0^T & b^T & c \end{array}\right], \quad \text{and} \quad x = \left[\begin{array}{c} \omega^0 \\ \hline \dot{\gamma} \\ \hline \dot{\eta} \end{array}\right]$$

and Hooker proves that the angular momentum of this system about its mass center is the product of the first row of A with x:

$$H = a_{00}\omega^0 + a_{0k}\dot{\gamma} + b_0\dot{\eta} \tag{B-1}$$

and that the 3 by 3 matrix a_{00} represents the instantaneous system inertia. The relation (B-1) is precisely that implemented in each of the subroutines MBDYFR, MBDYFN, and MBDYFL to calculate H (3 by 1). H is a 3 by 1 vector matrix whose elements are the components of the system angular momentum vector *in the reference body frame*. These three elements are available within the subroutine if the user wishes to extract them. He may also wish to transform them to an inertial reference frame in certain situations as a check on his simulation accuracy. However, the normal subroutine function as shown here in the examples and listings is to supply the user with only the *magnitude* of H, i.e.,

$$|H| = (h_1^2 + h_2^2 + h_3^2)^{\frac{1}{2}}$$

where

$$H = [h_1 h_2 h_3]^T$$

Appendix C

Subroutine MBDYFR Listing and User Requirements

Subroutine Entry Statements

CALL MBDYFR(NC, H, MB, MS, PB, PS, G, PI, NF, F,
ER, EI, SR, MF, RF, WF, ZF)

CALL MRATE(NC, TH, TB, TS, FB, FS, TF, FF, GM, GMD,
GMDD, DT, ET, WO, WDOT, DTD, ETD, HM)

Input / Output Variable Type and Storage Specifications

INTEGER NC, NF, H(n_c, 2), F(n_f, 3), PI(n + 1)

REAL MB(7), MS(n_c, 7), PB(n_c, 3), PS(n_c, n_c, 3), G(n, 3),
TH(n), TB(3), TS(n_c, 3), FB(3), FS(n_c, 3), GM(n),
GMD(n), GMDD(n), ER(n_f, 6n_k, N_k), EI(n_f, 6n_k, N_k),
MF(n_f, n_k, 7), RF(n_f, n_k, 3), WF(n_f, N_k), ZF(n_f, N_k),
TF(n_f, n_k, 3), FF(n_f, n_k, 3), DT(n_f, N_k), ET(n_f, N_k),
WO(3), SR(n_f, 3)

DOUBLE PRECISION WDOT(n + 3), DTD(n_f, N_k),
ETD(n_f, N_k)

External Subroutines Called

CHOLD—double precision subroutine for solving matrix equations of the form

$$Ax = B$$

where A is a square, symmetric, positive-definite matrix (see statement 1291).

Subroutine Setup

Insert the Fortran statement

PARAMETER QC = n_c, QH = n, QF = n_f, NK = n_k, NKT = N_k

(If more than one appendage is present, use the *largest* n_k and N_k for the PARAMETER statement to provide sufficient storage.)

Data Restrictions

$n > 1, n_f > 1, n_c > 1, n_k > 1, N_k > 1$

Core Storage Required

Code: 6500 words

Data: ~500 words (minimum; increases with n, n_f, etc.)

Listing

```
      SUBROUTINE MBDYFR(NC,C,MB,MA,PB,PA,G,PI,NF,F,ER,EI,SR,MF,RF,WF,ZF)
C
C     ADJUSTABLE DIMENSIONS
C
      INTEGER PI(1),C(NC,2)
      REAL MB(1),MA(NC,7),PB(NC,3),PA(NC,NC,3)
      PARAMETER QC=2,QH=3,QF=2,NK=1,NKT=2
      PARAMETER NAK=6*NK,S=QC+1,V=QH+3,V4=4*V,S3=3*S,Q=QH,NH=QH
      PARAMETER ST=V+2*QF*NKT,S4=4*ST
C
C     ADDITIONAL DIMENSIONED VARIABLES
C
      DOUBLE PRECISION A(ST,ST),BMASS(S)
      INTEGER EPS(Q,S),CPS(QC,S),H(Q),HI(S),FI(S),F(NF,3)
      REAL A00(3,3),AB(3,3),AOFI(QF,3,NKT),AOFR(QF,3,NKT),AKFR(QF,QH,NKT
     $),AKFI(QF,QH,NKT),AC(3,3),AS(Q,Q),AV(Q,3),AIS(3),B(QF,NK,3),BD(QF,
     $NK,3),CE(3),CL(3),CK(QF,3),CKD(QF,3),CDU(QF,3),CQ(3),CWWD(S,3),CV(
     $3),CW(S,3),DX(S,S),DY(S,S),DZ(S,S),DXO(S,S),DYO(S,S),DZO(S,S),DDSO
     $(QF,3),DLKR(QF,3,NKT),DLKI(QF,3,NKT),DLKRO(QF,3,NKT),DLKIO(QF,3,NK
     $T),DUR(3,NKT),DUI(3,NKT),DUXO(QF),DUYO(QF),DUZO(QF),EA(3),ER(NF,N6
     $K,NKT),EI(NF,N6K,NKT),FEXO(S),FEYO(S),FEZO(S),FS(S,3),GO(Q,3),GG(Q
     $,3),G(Q,3),GK(QF,3,NKT),GPSO(QF,3),GKOS(QF,3,NKT),I11,I22,I33,I12,
     $I13,I23,IXX(S),IYY(S),IZZ(S),IXY(S),IXZ(S),IYZ(S),LX(S,S),LY(S,S),
     $LZ(S,S),MSB(S),MS,MF(NF,NK,7),MCK(QF,3),MCKD(QF,3),PH(S,3,3),PSG(S
     $,S,3),PS(S,S,3,3),PK(QF,3,NKT),PGSO(QF,3),PSF(S,S,3,3),PKOS(QF,3,N
     $KT),RF(NF,NK,3),SR(QF,3),TXO(S),TYO(S),TZO(S),T(Q,3,3),TS(S,3),U(Q
     $F,NK,3),UD(QF,NK,3),VJ(3,3),VJD(3,3),VJDO(QF,3,3),VE(QF,3),VB(QF,N
     $6K),WF(NF,NKT),WWDE(QF,3),WGJ(QH,3),ZF(NF,NKT),ZSR(QF,NKT),ZSI(QF,
     $NKT),WW(3),HH(3)
      EQUIVALENCE (A,PS),(LX,DXO),(LY,DYO),(LZ,DZO)
      NB=NC+1
C
C     DEFINE EPS(K,J) USING C
C
      DO 86 K=1,NC
      DO 86 J=2,NB
      IF(K.EQ.(J-1)) CPS(K,J)=1
      IF(K.LT.(J-1)) GO TO 87
      GO TO 86
 87   CONTINUE
      J0=K+1
      J1=J-1
      DO 89 L=J0,J1
      IF(K.GT.(L-1)) GO TO 89
      IF((CPS(K,L).EQ.1).AND.(C(J-1,1).EQ.(L-1))) CPS(K,J)=1
 89   CONTINUE
 86   CONTINUE
      L=0
      DO 1 J=1,NC
      KK=C(J,2)
      DO 1 K=1,KK
      L=L+1
```

```
      DO 1 I=1,NB
    1 EPS(L,I)=CPS(J,I)
C
C     COMPUTE H(I)=C, WHERE I=HINGE LABEL AND C=CONNECTION LABEL
C
      I=0
      DO 8 J=2,NB
      KK=C(J-1,2)
      DO 8 K=1,KK
      I=I+1
    8 H(I)=J-1
C
C     COMPUTE HI(I)=J, WHERE I=BODY LABEL+1 AND J=NEAREST HINGE LABEL
C
      HI(1)=1
      HI(NB)=NH
      DO 47 I=NH,1
      IF(I.EQ.1) GO TO 47
      K1=H(I)
      K2=H(I-1)
      IF(K1.EQ.K2) GO TO 47
      HI(K2+1)=I-1
   47 CONTINUE
C
C     DEFINE FI(J)=K, WHERE J=BODY-LABEL+1 AND K IS APPENDAGE-LABEL
C                   (IF K=0, BODY HAS NO FLEX. APPENDAGE)
C
      DO 239 N=1,NB
  239 FI(N)=0
      DO 242 K=1,NF
      JN=F(K,1)+1
  242 FI(JN)=K
      NF=NF
      NB=NB
C
C     DEFINE SUBSTRUCTURE MASSES
C
      MSB(1)=MB(7)
      DO 248 N=2,NB
  248 MSB(N)=MA(N-1,7)
C
C     TOTAL NUMBER OF FLEX. APPENDAGE MODES TO BE RETAINED
C
      NTMO=0
      DO 461 K=1,NF
  461 NTMO=NTMO+F(K,3)
      NT2=2*NTMO
C
C     INITIAL CALCULATION OF BARYCENTER VECTORS W.R.T. BODY C.G.S
C                          AND HINGE POINTS
C
      IXX(1)=MB(1)
      IYY(1)=MB(2)
      IZZ(1)=MB(3)
      IXY(1)=MB(4)
      IXZ(1)=MB(5)
      IYZ(1)=MB(6)
      BMASS(1)=MB(7)
      TM=BMASS(1)
      DO 35 J=2,NB
      IXX(J)=MA(J-1,1)
      IYY(J)=MA(J-1,2)
      IZZ(J)=MA(J-1,3)
      IXY(J)=MA(J-1,4)
      IXZ(J)=MA(J-1,5)
      IYZ(J)=MA(J-1,6)
      BMASS(J)=MA(J-1,7)
   35 TM=TM+BMASS(J)
      DO 149 I=1,NB
```

```
      I1=I-1
      DO 149 J=1,NB
      J1=J-1
      IF(I.EQ.J) GO TO 163
      IF(I.GT.J) GO TO 70
      IF(I.EQ.1) GO TO 80
      IF(CPS(I1,J).EQ.1) GO TO 400
70    LX(I,J)=PA(I1,I1,1)
      LY(I,J)=PA(I1,I1,2)
      LZ(I,J)=PA(I1,I1,3)
      GO TO 149
400   CONTINUE
      DO 600 K=1,J1
      IF(CPS(K,J).EQ.1) GO TO 500
600   CONTINUE
      GO TO 149
500   LX(I,J)=PA(I1,K,1)
      LY(I,J)=PA(I1,K,2)
      LZ(I,J)=PA(I1,K,3)
      GO TO 149
80    DO 90 L=1,J1
      IF(CPS(L,J).EQ.1) GO TO 101
90    CONTINUE
      GO TO 149
101   LX(I,J)=PB(L,1)
      LY(I,J)=PB(L,2)
      LZ(I,J)=PB(L,3)
      GO TO 149
163   LX(I,J)=0.
      LY(I,J)=0.
      LZ(I,J)=0.
149   CONTINUE
      DO 13 N=1,NB
      DO 13 J=1,NB
      DX(N,J)=LX(N,J)
      DY(N,J)=LY(N,J)
      DZ(N,J)=LZ(N,J)
      DO 13 K=1,NB
      DX(N,J)=DX(N,J)-(BMASS(K)/TM)*LX(N,K)
      DY(N,J)=DY(N,J)-(BMASS(K)/TM)*LY(N,K)
13    DZ(N,J)=DZ(N,J)-(BMASS(K)/TM)*LZ(N,K)
C
C     NOMINAL SPIN RATE CENTRIFUGAL FORCES
C
      DO 736 K=1,NF
      I=F(K,1)+1
      R1=SR(K,1)
      R2=SR(K,2)
      R3=SR(K,3)
      D1=DX(I,I)
      D2=DY(I,I)
      D3=DZ(I,I)
      WWDE(K,1)=-R3*(R3*D1-R1*D3)+R2*(-R2*D1+R1*D2)
      WWDE(K,2)=R3*(-R3*D2+R2*D3)-R1*(R1*D2-R2*D1)
736   WWDE(K,3)=-R2*(-R3*D2+R2*D3)+R1*(R3*D1-R1*D3)
C
C     CALCULATION OF AUGMENTED INERTIA DYADICS FOR EACH BODY
C
      DO 31 N=1,NB
      PH(N,1,1)=IXX(N)
      PH(N,1,2)=-IXY(N)
      PH(N,1,3)=-IXZ(N)
      PH(N,2,2)=IYY(N)
      PH(N,2,3)=-IYZ(N)
      PH(N,3,3)=IZZ(N)
      DO 30 J=1,NB
      PH(N,1,1)=PH(N,1,1)+BMASS(J)*(DY(N,J)**2+DZ(N,J)**2)
      PH(N,1,2)=PH(N,1,2)-BMASS(J)*DX(N,J)*DY(N,J)
      PH(N,1,3)=PH(N,1,3)-BMASS(J)*DX(N,J)*DZ(N,J)
```

```
          PH(N,2,2)=PH(N,2,2)+BMASS(J)*(DX(N,J)**2+DZ(N,J)**2)
          PH(N,2,3)=PH(N,2,3)-BMASS(J)*DY(N,J)*DZ(N,J)
   30     PH(N,3,3)=PH(N,3,3)+BMASS(J)*(DX(N,J)**2+DY(N,J)**2)
          PH(N,2,1)=PH(N,1,2)
          PH(N,3,1)=PH(N,1,3)
   31     PH(N,3,2)=PH(N,2,3)
C
C         COMPUTE PK AND GK (3 X NKT ARRAYS)
C
          DO 201 K=1,NF
          LN=F(K,2)
          JNT=F(K,3)
          DO 201 I=1,3
          DO 201 J=1,JNT
          PK(K,I,J)=0.
          GK(K,I,J)=0.
          DO 202 L=1,LN
          LL=6*(L-1)+1
          PK(K,I,J)=PK(K,I,J)+MF(K,L,7)*ER(K,LL,J)
  202     GK(K,I,J)=GK(K,I,J)+MF(K,L,7)*EI(K,LL,J)
          PK(K,I,J)=2.*PK(K,I,J)
          GK(K,I,J)=2.*GK(K,I,J)
  201     CONTINUE
C
C         COMPUTE DLKR-AND DLKI-TRANSPOSE MATRICES (3 X NKT ARRAYS)
C
          DO 203 K=1,NF
          LN=F(K,2)
          JNT=F(K,3)
          DO 203 J=1,JNT
          DO 204 I=1,3
          DLKR(K,I,J)=0.
  204     DLKI(K,I,J)=0.
          DO 205 L=1,LN
          L1=6*(L-1)+1
          L2=L1+1
          L3=L2+1
          L4=L3+1
          L5=L4+1
          L6=L5+1
          DLKR(K,1,J)=DLKR(K,1,J)+MF(K,L,7)*(EI(K,L3,J)*RF(K,L,2)
         S     -EI(K,L2,J)*RF(K,L,3))+MF(K,L,1)*EI(K,L4,J)
         S     -MF(K,L,4)*EI(K,L5,J) - MF(K,L,5)*EI(K,L6,J)
          DLKR(K,2,J)=DLKR(K,2,J)+MF(K,L,7)*(EI(K,L1,J)*RF(K,L,3)
         S     -EI(K,L3,J)*RF(K,L,1))+MF(K,L,2)*EI(K,L5,J)
         S     -MF(K,L,4)*EI(K,L4,J) - MF(K,L,6)*EI(K,L6,J)
          DLKR(K,3,J)=DLKR(K,3,J)+MF(K,L,7)*(EI(K,L2,J)*RF(K,L,1)
         S     -EI(K,L1,J)*RF(K,L,2))+MF(K,L,3)*EI(K,L6,J)
         S     -MF(K,L,5)*EI(K,L4,J) - MF(K,L,6)*EI(K,L5,J)
          DLKI(K,1,J)=DLKI(K,1,J)+MF(K,L,7)*(ER(K,L3,J)*RF(K,L,2)
         S     -ER(K,L2,J)*RF(K,L,3))+MF(K,L,1)*ER(K,L4,J)
         S     -MF(K,L,4)*ER(K,L5,J) - MF(K,L,5)*ER(K,L6,J)
          DLKI(K,2,J)=DLKI(K,2,J)+MF(K,L,7)*(ER(K,L1,J)*RF(K,L,3)
         S     -ER(K,L3,J)*RF(K,L,1))+MF(K,L,2)*ER(K,L5,J)
         S     -MF(K,L,4)*ER(K,L4,J) - MF(K,L,6)*ER(K,L6,J)
          DLKI(K,3,J)=DLKI(K,3,J)+MF(K,L,7)*(ER(K,L2,J)*RF(K,L,1)
         S     -ER(K,L1,J)*RF(K,L,2))+MF(K,L,3)*ER(K,L6,J)
         S     -MF(K,L,5)*ER(K,L4,J) - MF(K,L,6)*ER(K,L5,J)
  205     CONTINUE
          DO 206 I=1,3
          DLKR(K,I,J)=-2.*DLKR(K,I,J)*WF(K,J)
  206     DLKI(K,I,J)=-2.*DLKI(K,I,J)*WF(K,J)
  203     CONTINUE
          RETURN
          ENTRY MRATE(NC,TH,TB,TA,FB,FA,TF,FF,GM,GMD,GMDD,DT,ET,WO,WDOT,
         SDTD,ETD,HM,U,UD)
          REAL TF(QF,NK,3),FF(QF,NK,3),DT(QF,NKT),ET(QF,NKT),TB(3),TA(NC,3),
         SFB(3),FA(NC,3),GM(1),GMD(1),GMDD(1),TH(1),WO(3),WXO(S),WYO(S),WZO(
         SS),E(S3,1)
```

```
      DOUBLE PRECISION EC(ST),DTD(QF,NKT),ETD(QF,NKT),WDOT(V)
C
C     BODY-TO-BODY COORDINATE TRANSFORMATION MATRICES
C
      DO 335 J=1,NH
      MM=J-1
      N=H(J)+1
      SGM=SIN(GM(J))
      CGM=COS(GM(J))
      CGM1=1.-CGM
      G1=CGM1*G(J,1)
      G2=CGM1*G(J,2)
      G3=CGM1*G(J,3)
      SG1=SGM*G(J,1)
      SG2=SGM*G(J,2)
      SG3=SGM*G(J,3)
      G1S=G1*G(J,1)
      G2S=G2*G(J,2)
      G3S=G3*G(J,3)
      G12=G1*G(J,2)
      G13=G1*G(J,3)
      G23=G2*G(J,3)
      AB(1,1)=CGM+G1S
      AB(1,2)=SG3+G12
      AB(1,3)=-SG2+G13
      AB(2,1)=-SG3+G12
      AB(2,2)=CGM+G2S
      AB(2,3)=SG1+G23
      AB(3,1)=SG2+G13
      AB(3,2)=-SG1+G23
      AB(3,3)=CGM+G3S
      IF(J.EQ.1) GO TO 3350
      DO 321 L=MM,1
      IF(EPS(L,N).EQ.1) GO TO 322
 321  CONTINUE
      GO TO 3350
 322  K=L
      DO 334 L=1,3
      DO 334 M=1,3
      T(J,L,M)=0.
      DO 334 I=1,3
 334  T(J,L,M)=T(J,L,M)+AB(L,I)*T(K,I,M)
      GO TO 335
 3350 CONTINUE
      DO 3351 L=1,3
      DO 3351 M=1,3
 3351 T(J,L,M)=AB(L,M)
 335  CONTINUE
C
C     COORD. TRANSFORMATION OF G VECTORS (TO REF. BODY FRAME)
C
      DO 362 I=1,NH
      DO 362 J=1,3
      GO(I,J)=0.
      DO 362 K=1,3
      GO(I,J)=GO(I,J)+T(I,K,J)*G(I,K)
 362  CONTINUE
C
C     ANG. VELOCITY COMPONENTS OF EACH BODY (IN REF. BODY FRAME)
C
      DO 366 K=1,NH
      GG(K,1)=GMD(K)*GO(K,1)
      GG(K,2)=GMD(K)*GO(K,2)
 366  GG(K,3)=GMD(K)*GO(K,3)
      DO 361 J=1,NB
      KV=HI(J)
      WXO(J)=WO(1)
      WYO(J)=WO(2)
      WZO(J)=WO(3)
```

```
      DO 36 K=1,KV
      IF(EPS(K,J).EQ.0) GO TO 36
      WXO(J)=WXO(J)+GG(K,1)
      WYO(J)=WYO(J)+GG(K,2)
      WZO(J)=WZO(J)+GG(K,3)
36    CONTINUE
361   CONTINUE
C
C     ANG. VELOCITY COMPONENTS AT EACH HINGE (IN REF. BODY FRAME)
C
      DO 3666 M=1,NH
      M1=M+1
      MC=H(M)+1
      NI=HI(MC)
      WHXO=WXO(MC)
      WHYO=WYO(MC)
      WHZO=WZO(MC)
      IF(NI.EQ.M) GO TO 3667
      DO 3668 N=M1,NI
      WHXO=WHXO-GG(N,1)
      WHYO=WHYO-GG(N,2)
3668  WHZO=WHZO-GG(N,3)
3667  CONTINUE
      WGJ(M,1)=GG(M,3)*WHYO-GG(M,2)*WHZO
      WGJ(M,2)=GG(M,1)*WHZO-GG(M,3)*WHXO
      WGJ(M,3)=GG(M,2)*WHXO-GG(M,1)*WHYO
3666  CONTINUE
C
C     TRANSFORM PK AND GK MATRICES TO REFERENCE BODY BASIS-MULT.BY FREQ.
C
      DO 468 K=1,NF
      KK=F(K,1)+1
      JNT=F(K,3)
      IF(KK.EQ.1) GO TO 4720
      M=HI(KK)
      DO 472 I=1,3
      DO 472 J=1,JNT
      DLKRO(K,I,J)=0.
      DLKIO(K,I,J)=0.
      PKOS(K,I,J)=0.
      GKOS(K,I,J)=0.
      DO 469 L=1,3
      DLKRO(K,I,J)=DLKRO(K,I,J)+T(M,L,I)*DLKR(K,L,J)
      DLKIO(K,I,J)=DLKIO(K,I,J)+T(M,L,I)*DLKI(K,L,J)
      PKOS(K,I,J)=PKOS(K,I,J)+T(M,L,I)*PK(K,L,J)
469   GKOS(K,I,J)=GKOS(K,I,J)+T(M,L,I)*GK(K,L,J)
      PKOS(K,I,J)=PKOS(K,I,J)*WF(K,J)
472   GKOS(K,I,J)=GKOS(K,I,J)*WF(K,J)
      GO TO 468
4720  CONTINUE
      DO 4721 I=1,3
      DO 4721 J=1,JNT
      DLKRO(K,I,J)=DLKR(K,I,J)
      DLKIO(K,I,J)=DLKI(K,I,J)
      PKOS(K,I,J)=PK(K,I,J)*WF(K,J)
4721  GKOS(K,I,J)=GK(K,I,J)*WF(K,J)
468   CONTINUE
C
C     COMPUTE TOTAL EXTERNAL FORCE ON EACH SUBSTRUCTURE (IN REF. COORD.)
C
      FEXO(1)=FB(1)
      FEYO(1)=FB(2)
      FEZO(1)=FB(3)
      IF(FI(1).EQ.0) GO TO 254
      IL=FI(1)
      JN=F(IL,2)
      DO 253 J=1,JN
      FEXO(1)=FEXO(1)+FF(IL,J,1)
      FEYO(1)=FEYO(1)+FF(IL,J,2)
```

```
 253   FEZO(I)=FEZO(I)+FF(IL,J,3)
 254   CONTINUE
       FS(1,1)=FEXO(1)
       FS(1,2)=FEYO(1)
       FS(1,3)=FEZO(1)
       DO 246 N=2,NB
       K=N-1
       DO 2460 L=1,3
 2460  FS(N,L)=FA(K,L)
       IF(FI(N).EQ.0) GO TO 246
       IL=FI(N)
       JN=F(IL,2)
       DO 245 J=1,JN
       DO 245 I=1,3
 245   FS(N,I)=FS(N,I)+FF(IL,J,I)
 246   CONTINUE
 C
 C     COMPUTE TRANSL. AND ROTAT. DISPLACEMENTS OF APPENDAGE SUB-BODIES
 C
       DO 232 K=1,NF
       JN=F(K,2)
       LK=F(K,3)
       DO 233 J=1,JN
       DO 233 I=1,3
       U(K,J,I)=0.
       B(K,J,I)=0.
       UD(K,J,I)=0.
       BD(K,J,I)=0.
       ID=(J-1)*6+1
       IR=ID+3
       DO 233 L=1,LK
       U(K,J,I)=U(K,J,I)+2.*ER(K,ID,L)*DT(K,L)-2.*EI(K,ID,L)*ET(K,L)
       B(K,J,I)=B(K,J,I)+2.*ER(K,IR,L)*DT(K,L)-2.*EI(K,IR,L)*ET(K,L)
       UD(K,J,I)=UD(K,J,I)-2.*ER(K,ID,L)*ET(K,L)*WF(K,L)
      S                        -2.*EI(K,ID,L)*DT(K,L)*WF(K,L)
 233   BD(K,J,I)=BD(K,J,I)-2.*ER(K,IR,L)*ET(K,L)*WF(K,L)
      S                        -2.*EI(K,IR,L)*DT(K,L)*WF(K,L)
 232   CONTINUE
 C
 C     COMPUTE C.M. PERTURBATION (FROM NOM. UNDEFORMED LOCATION) ON EACH
 C              SUBSTRUCTURE WITH AN APPENDAGE (LOCAL COORDS.)
 C
       DO 262 K=1,NF
       IK=F(K,1)+1
       JN=F(K,2)
       DO 263 I=1,3
       MCKD(K,I)=0.
 263   MCK(K,I)=0.
       DO 265 J=1,JN
       DO 265 I=1,3
       MCKD(K,I)=MCKD(K,I)-UD(K,J,I)*MF(K,J,7)
 265   MCK(K,I)=MCK(K,I)-U(K,J,I)*MF(K,J,7)
       DO 266 I=1,3
       CKD(K,I)=MCKD(K,I)/MSB(IK)
 266   CK(K,I)=MCK(K,I)/MSB(IK)
 262   CONTINUE
 C
 C     COMPUTE TOTAL EXTERNAL TORQUE ON EACH SUBSTRUCTURE W.R.T. ITS
 C              INSTANTANEOUS C.M. (IN LOCAL COORD.)
 C
       DO 268 L=1,3
 268   TS(1,L)=TB(L)
       DO 267 N=2,NB
       K=N-1
       DO 267 L=1,3
 267   TS(N,L)=TA(K,L)
       DO 2670 N=1,NB
       IL=FI(N)
       IF(IL.EQ.0) GO TO 2670
```

```
      JN=F(IL,2)
      DO 2671 J=1,JN
      DO 2671 L=1,3
 2671 TS(N,L)=TS(N,L)+TF(IL,J,L)
 2670 CONTINUE
      DO 269 N=1,NB
      K=FI(N)
      IF(K.EQ.0) GO TO 269
      TS(N,1)=TS(N,1)+CK(K,2)*FS(N,3)-CK(K,3)*FS(N,2)
      TS(N,2)=TS(N,2)+CK(K,3)*FS(N,1)-CK(K,1)*FS(N,3)
      TS(N,3)=TS(N,3)+CK(K,1)*FS(N,2)-CK(K,2)*FS(N,1)
  269 CONTINUE
      DO 271 N=1,NB
      K=FI(N)
      IF(K.EQ.0) GO TO 271
      JN=F(K,2)
      DO 272 J=1,JN
      RUX=RF(K,J,1)+U(K,J,1)
      RUY=RF(K,J,2)+U(K,J,2)
      RUZ=RF(K,J,3)+U(K,J,3)
      TS(N,1)=TS(N,1)+RUY*FF(K,J,3)-RUZ*FF(K,J,2)
      TS(N,2)=TS(N,2)+RUZ*FF(K,J,1)-RUX*FF(K,J,3)
  272 TS(N,3)=TS(N,3)+RUX*FF(K,J,2)-RUY*FF(K,J,1)
  271 CONTINUE
C
C     TRANSFORM VECTORS TO REF. BODY FRAME
C
      TXO(1)=TS(1,1)
      TYO(1)=TS(1,2)
      TZO(1)=TS(1,3)
      DO 17 I=2,NB
      M=HI(I)
      K=I-1
      L=C(K,1)+1
      FEXO(I)=T(M,1,1)*FS(I,1)+T(M,2,1)*FS(I,2)+T(M,3,1)*FS(I,3)
      FEYO(I)=T(M,1,2)*FS(I,1)+T(M,2,2)*FS(I,2)+T(M,3,2)*FS(I,3)
      FEZO(I)=T(M,1,3)*FS(I,1)+T(M,2,3)*FS(I,2)+T(M,3,3)*FS(I,3)
      TXO(I) =T(M,1,1)*TS(I,1)+T(M,2,1)*TS(I,2)+T(M,3,1)*TS(I,3)
      TYO(I) =T(M,1,2)*TS(I,1)+T(M,2,2)*TS(I,2)+T(M,3,2)*TS(I,3)
      TZO(I) =T(M,1,3)*TS(I,1)+T(M,2,3)*TS(I,2)+T(M,3,3)*TS(I,3)
      DXO(I,I)=T(M,1,1)*DX(I,I)+T(M,2,1)*DY(I,I)+T(M,3,1)*DZ(I,I)
      DYO(I,I)=T(M,1,2)*DX(I,I)+T(M,2,2)*DY(I,I)+T(M,3,2)*DZ(I,I)
      DZO(I,I)=T(M,1,3)*DX(I,I)+T(M,2,3)*DY(I,I)+T(M,3,3)*DZ(I,I)
      DXO(I,L)=T(M,1,1)*DX(I,L)+T(M,2,1)*DY(I,L)+T(M,3,1)*DZ(I,L)
      DYO(I,L)=T(M,1,2)*DX(I,L)+T(M,2,2)*DY(I,L)+T(M,3,2)*DZ(I,L)
      DZO(I,L)=T(M,1,3)*DX(I,L)+T(M,2,3)*DY(I,L)+T(M,3,3)*DZ(I,L)
      DO 17 J=1,NB
      IF(I.EQ.J) GO TO 17
      IF(CPS(K,J).EQ.1) GO TO 177
      IF(C(K,1).EQ.(J-1)) GO TO 17
      DXO(I,J)=DXO(I,L)
      DYO(I,J)=DYO(I,L)
      DZO(I,J)=DZO(I,L)
      GO TO 17
  177 DXO(I,J)=T(M,1,1)*DX(I,J)+T(M,2,1)*DY(I,J)+T(M,3,1)*DZ(I,J)
      DYO(I,J)=T(M,1,2)*DX(I,J)+T(M,2,2)*DY(I,J)+T(M,3,2)*DZ(I,J)
      DZO(I,J)=T(M,1,3)*DX(I,J)+T(M,2,3)*DY(I,J)+T(M,3,3)*DZ(I,J)
   17 CONTINUE
      DO 367 I=1,NB
      DXO(I,I)=DX(I,I)
      DYO(I,I)=DY(I,I)
  367 DZO(I,I)=DZ(I,I)
C
C     COMPUTE TOTAL EXTERNAL FORCE ON VEHICLE (IN REF. COORD.)
C
      FTXO=0.
      FTYO=0.
      FTZO=0.
      DO 247 N=1,NB
```

```
      FTXO=FTXO+FEXO(N)
      FTYO=FTYO+FEYO(N)
  247 FTZO=FTZO+FEZO(N)
C
C     ADDITIONAL AUGMENTED INERTIA DYADICS (IN REF.BODY FRAME)
C
      DO 37 I=1,NB
      DO 37 J=1,NB
      IF(I.GE.J) GO TO 37
      DX2=DXO(I,J)*DXO(J,I)
      DY2=DYO(I,J)*DYO(J,I)
      DZ2=DZO(I,J)*DZO(J,I)
      PS(I,J,1,1)=-TM*(DY2+DZ2)
      PS(I,J,1,2)=TM*DXO(J,I)*DYO(I,J)
      PS(I,J,1,3)=TM*DXO(J,I)*DZO(I,J)
      PS(I,J,2,1)=TM*DYO(J,I)*DXO(I,J)
      PS(I,J,2,2)=-TM*(DX2+DZ2)
      PS(I,J,2,3)=TM*DYO(J,I)*DZO(I,J)
      PS(I,J,3,1)=TM*DZO(J,I)*DXO(I,J)
      PS(I,J,3,2)=TM*DZO(J,I)*DYO(I,J)
      PS(I,J,3,3)=-TM*(DX2+DY2)
      DO 378 M=1,3
      DO 378 N=1,3
  378 PS(J,I,M,N)=PS(I,J,N,M)
   37 CONTINUE
      DO 751 J=1,NB
      DO 751 M=1,3
      DO 751 N=1,3
  751 PS(J,J,M,N)=PH(J,M,N)
C
C     COMPUTE VARIABLE PART OF APPENDAGE INERTIA (IN SUBSTR. COORDS.)
C
      DO 236 K=1,NF
      KK=F(K,1)+1
      M=HI(KK)
      JN=F(K,2)
      DO 235 I=1,3
      DO 235 J=1,3
      VJ(I,J)=0.
  235 VJD(I,J)=0.
      DO 234 J=1,JN
      I11=MF(K,J,1)
      I22=MF(K,J,2)
      I33=MF(K,J,3)
      I12=-MF(K,J,4)
      I13=-MF(K,J,5)
      I23=-MF(K,J,6)
      MS=MF(K,J,7)
      R1=RF(K,J,1)
      R2=RF(K,J,2)
      R3=RF(K,J,3)
      U1=U(K,J,1)
      U2=U(K,J,2)
      U3=U(K,J,3)
      B1=B(K,J,1)
      B2=B(K,J,2)
      B3=B(K,J,3)
      VJ(1,1)=VJ(1,1)+2.*(MS*(R2*U2+R3*U3)-I12*B3+I13*B2)
      VJ(2,2)=VJ(2,2)+2.*(MS*(R1*U1+R3*U3)-I23*B1+I12*B3)
      VJ(3,3)=VJ(3,3)+2.*(MS*(R1*U1+R2*U2)-I13*B2+I23*B1)
      VJ(1,2)=VJ(1,2)-MS*(R1*U2+R2*U1)-I13*B1+I23*B2-B3*(I22-I11)
      VJ(1,3)=VJ(1,3)-MS*(R1*U3+R3*U1)+I12*B1-I23*B3-B2*(I11-I33)
      VJ(2,3)=VJ(2,3)-MS*(R2*U3+R3*U2)-I12*B2+I13*B3-B1*(I33-I22)
      U1=UD(K,J,1)
      U2=UD(K,J,2)
      U3=UD(K,J,3)
      B1=BD(K,J,1)
      B2=BD(K,J,2)
      B3=BD(K,J,3)
```

```
      VJD(1,1)=VJD(1,1)+2.*(MS*(R2*U2+R3*U3)-I12*B3+I13*B2)
      VJD(2,2)=VJD(2,2)+2.*(MS*(R1*U1+R3*U3)-I23*B1+I12*B3)
      VJD(3,3)=VJD(3,3)+2.*(MS*(R1*U1+R2*U2)-I13*B2+I23*B1)
      VJD(1,2)=VJD(1,2)-MS*(R1*U2+R2*U1)-I13*B1+I23*B2-B3*(I22-I11)
      VJD(1,3)=VJD(1,3)-MS*(R1*U3+R3*U1)+I12*B1-I23*B3-B2*(I11-I33)
  234 VJD(2,3)=VJD(2,3)-MS*(R2*U3+R3*U2)-I12*B2+I13*B3-B1*(I33-I22)
      VJ(2,1)=VJ(1,2)
      VJ(3,1)=VJ(1,3)
      VJ(3,2)=VJ(2,3)
      DO 495 I=1,3
      DO 495 J=1,3
  495 PS(KK,KK,I,J)=PS(KK,KK,I,J)+VJ(I,J)
      VJD(2,1)=VJD(1,2)
      VJD(3,1)=VJD(1,3)
      VJD(3,2)=VJD(2,3)
C
C     CONVERT INERTIA MATRIX TO REF. BODY COORDS.
C
      IF(KK.EQ.1) GO TO 2370
      DO 237 J=1,3
      DO 237 I=1,3
      AC(J,I)=0.
      DO 237 L=1,3
      AC(J,I)=AC(J,I)+VJD(J,L)*T(M,L,I)
  237 CONTINUE
      DO 238 J=1,3
      DO 238 I=1,3
      VJDO(K,J,I)=0.
      DO 238 L=1,3
      VJDO(K,J,I)=VJDO(K,J,I)+T(M,L,J)*AC(L,I)
  238 CONTINUE
      GO TO 236
 2370 CONTINUE
      DO 2371 J=1,3
      DO 2371 I=1,3
 2371 VJDO(K,J,I)=VJD(J,I)
  236 CONTINUE
C
C     TRANSFORM AUGMENTED BODY INERTIA DYADICS TO REF. BODY FRAME
C
      DO 363 I=2,NB
      M=HI(I)
      DO 364 J=1,3
      DO 364 K=1,3
      AB(J,K)=0.
      DO 364 L=1,3
      AB(J,K)=AB(J,K)+PS(I,I,J,L)*T(M,L,K)
  364 CONTINUE
      DO 365 J=1,3
      DO 365 K=1,3
      PS(I,I,J,K)=0.
      DO 365 L=1,3
      PS(I,I,J,K)=PS(I,I,J,K)+T(M,L,J)*AB(L,K)
  365 CONTINUE
  363 CONTINUE
C
C     COMPUTE THE PGSO,GPSO, AND DDSO VECTORS FOR EACH FLEX. APPEND.
C
      DO 208 K=1,NF
      KK=F(K,1)+1
      M=HI(KK)
      JNT=F(K,3)
      DO 207 I=1,3
      CV(I)=0.
      DO 207 J=1,JNT
  207 CV(I)=CV(I)+DLKR(K,I,J)*DT(K,J)+DLKI(K,I,J)*ET(K,J)
      IF(KK.EQ.1) GO TO 2090
      DO 209 I=1,3
      PGSO(K,I)=0.
```

ORIGINAL PAGE IS OF POOR QUALITY

C, 2

```
      GPSO(K,I)=0.
      DDSO(K,I)=0.
      DO 209 J=1,3
      PGSO(K,I)=PGSO(K,I)+T(M,J,I)*(-MCK(K,J))
      GPSO(K,I)=GPSO(K,I)+T(M,J,I)*(-MCKD(K,J))
209   DDSO(K,I)=DDSO(K,I)+T(M,J,I)*CV(J)
      GO TO 208
2090  CONTINUE
      DO 2091 I=1,3
      PGSO(K,I)=-MCK(K,I)
      GPSO(K,I)=-MCKD(K,I)
2091  DDSO(K,I)=CV(I)
208   CONTINUE
C
C     VECTOR CROSS PRODUCTS DESCRIBING SYSTEM ROTATIONAL COUPLING.
C       (QUADRATIC TERMS INVOLVING THE CONNECTING BODY ANGULAR
C       VELOCITIES AND THE MUTUAL BARYCENTER-HINGE VECTORS)
C
      DO 261 K=1,NF
      I=F(K,1)+1
      DUX=WZO(I)*PGSO(K,2)-WYO(I)*PGSO(K,3)
      DUY=WXO(I)*PGSO(K,3)-WZO(I)*PGSO(K,1)
      DUZ=WYO(I)*PGSO(K,1)-WXO(I)*PGSO(K,2)
      DUXO(K)=WYO(I)*(DUZ-2.*GPSO(K,3))-WZO(I)*(DUY-2.*GPSO(K,2))
      DUYO(K)=WZO(I)*(DUX-2.*GPSO(K,1))-WXO(I)*(DUZ-2.*GPSO(K,3))
261   DUZO(K)=WXO(I)*(DUY-2.*GPSO(K,2))-WYO(I)*(DUX-2.*GPSO(K,1))
      DO 230 N=1,NB
      I=FI(N)
      DO 476 J=1,3
476   CWWD(N,J)=0.
      CPX=0.
      CPY=0.
      CPZ=0.
      CPFX=0.
      CPFY=0.
      CPFZ=0.
      DCPX=0.
      DCPY=0.
      DCPZ=0.
      DO 2301 L=1,NB
      IL=FI(L)
      IF(IL.EQ.0) GO TO 2303
      DCPX=DCPX+DYO(N,L)*DUZO(IL)-DZO(N,L)*DUYO(IL)
      DCPY=DCPY+DZO(N,L)*DUXO(IL)-DXO(N,L)*DUZO(IL)
      DCPZ=DCPZ+DXO(N,L)*DUYO(IL)-DYO(N,L)*DUXO(IL)
2303  CONTINUE
      WDX=WYO(L)*DZO(L,N)-WZO(L)*DYO(L,N)
      WDY=WZO(L)*DXO(L,N)-WXO(L)*DZO(L,N)
      WDZ=WXO(L)*DYO(L,N)-WYO(L)*DXO(L,N)
      WWFDX=WYO(L)*WDZ-WZO(L)*WDY
      WWFDY=WZO(L)*WDX-WXO(L)*WDZ
      WWFDZ=WXO(L)*WDY-WYO(L)*WDX
      IF(I.EQ.0) GO TO 482
      CWWD(N,1)=CWWD(N,1)+WWFDX
      CWWD(N,2)=CWWD(N,2)+WWFDY
      CWWD(N,3)=CWWD(N,3)+WWFDZ
482   CONTINUE
      CPFX=CPFX+WWFDX
      CPFY=CPFY+WWFDY
      CPFZ=CPFZ+WWFDZ
      IF(N.EQ.L) GO TO 2301
      WWDX=TM*WWFDX+FEXO(L)
      WWDY=TM*WWFDY+FEYO(L)
      WWDZ=TM*WWFDZ+FEZO(L)
      DWWDX=DYO(N,L)*WWDZ-DZO(N,L)*WWDY
      DWWDY=DZO(N,L)*WWDX-DXO(N,L)*WWDZ
      DWWDZ=DXO(N,L)*WWDY-DYO(N,L)*WWDX
      CPX=CPX+DWWDX
      CPY=CPY+DWWDY
```

```
      CPZ=CPZ+DWWDZ
2301  CONTINUE
      DFX=DYO(N,N)*FEZO(N)-DZO(N,N)*FEYO(N)
      DFY=DZO(N,N)*FEXO(N)-DXO(N,N)*FEZO(N)
      DFZ=DXO(N,N)*FEYO(N)-DYO(N,N)*FEXO(N)
      HX=PS(N,N,1,1)*WXO(N)+PS(N,N,1,2)*WYO(N)+PS(N,N,1,3)*WZO(N)
      HY=PS(N,N,2,1)*WXO(N)+PS(N,N,2,2)*WYO(N)+PS(N,N,2,3)*WZO(N)
      HZ=PS(N,N,3,1)*WXO(N)+PS(N,N,3,2)*WYO(N)+PS(N,N,3,3)*WZO(N)
      IF(I.EQ.0) GO TO 243
      HXD=VJDO(I,1,1)*WXO(N)+VJDO(I,1,2)*WYO(N)+VJDO(I,1,3)*WZO(N)
      HYD=VJDO(I,2,1)*WXO(N)+VJDO(I,2,2)*WYO(N)+VJDO(I,2,3)*WZO(N)
      HZD=VJDO(I,3,1)*WXO(N)+VJDO(I,3,2)*WYO(N)+VJDO(I,3,3)*WZO(N)
      FACT=MSB(N)/TM
      FTXM=FTXO*FACT
      FTYM=FTYO*FACT
      FTZM=FTZO*FACT
      PGFX=(PGSO(I,2)*(FEZO(N)-FTZM)-PGSO(I,3)*(FEYO(N)-FTYM))/MSB(N)
      PGFY=(PGSO(I,3)*(FEXO(N)-FTXM)-PGSO(I,1)*(FEZO(N)-FTZM))/MSB(N)
      PGFZ=(PGSO(I,1)*(FEYO(N)-FTYM)-PGSO(I,2)*(FEXO(N)-FTXM))/MSB(N)
      PWWDX=PGSO(I,2)*CPFZ-PGSO(I,3)*CPFY
      PWWDY=PGSO(I,3)*CPFX-PGSO(I,1)*CPFZ
      PWWDZ=PGSO(I,1)*CPFY-PGSO(I,2)*CPFX
      WDDSXO=WYO(N)*DDSO(I,3)-WZO(N)*DDSO(I,2)
      WDDSYO=WZO(N)*DDSO(I,1)-WXO(N)*DDSO(I,3)
      WDDSZO=WXO(N)*DDSO(I,2)-WYO(N)*DDSO(I,1)
      GO TO 244
243   CONTINUE
      HXD=0.
      HYD=0.
      HZD=0.
      PGFX=0.
      PGFY=0.
      PGFZ=0.
      PWWDX=0.
      PWWDY=0.
      PWWDZ=0.
      WDDSXO=0.
      WDDSYO=0.
      WDDSZO=0.
244   CONTINUE
      K = 3*(N-1)
      E(K+1,1)=HY*WZO(N)-HZ*WYO(N)+TXO(N)+CPX+DFX-HXD+PGFX-PWWDX-WDDSXO
     $+DCPX
      E(K+2,1)=HZ*WXO(N)-HX*WZO(N)+TYO(N)+CPY+DFY-HYD+PGFY-PWWDY-WDDSYO
     $+DCPY
      E(K+3,1)=HX*WYO(N)-HY*WXO(N)+TZO(N)+CPZ+DFZ-HZD+PGFZ-PWWDZ-WDDSZO
     $+DCPZ
230   CONTINUE
C
C     AOO MATRIX ELEMENT COMPUTATION (3X3)
C
      DO 3001 I=1,3
      DO 3001 J=1,3
3001  AOO(I,J)=0.
      DO 3 I=1,NB
      DO 3 J=1,NB
      AOO(1,1)=AOO(1,1)+PS(I,J,1,1)
      AOO(1,2)=AOO(1,2)+PS(I,J,1,2)
      AOO(1,3)=AOO(1,3)+PS(I,J,1,3)
      AOO(2,2)=AOO(2,2)+PS(I,J,2,2)
      AOO(2,3)=AOO(2,3)+PS(I,J,2,3)
      AOO(3,3)=AOO(3,3)+PS(I,J,3,3)
3     CONTINUE
      AOO(2,1)=AOO(1,2)
      AOO(3,1)=AOO(1,3)
      AOO(3,2)=AOO(2,3)
C
C     FLEX. APPEND. CONTRIBUTION TO AOO MATRIX COMPUTATION (3X3)
C
```

```
      DO 210 K=1,NB
      KK=FI(K)
      DO 210 L=1,NB
      IF(K.GT.L) GO TO 210
      DO 2103 I=1,3
      DO 2103 J=1,3
2103  PSF(K,L,I,J)=0.
      LL=FI(L)
      IF(KK.EQ.0) GO TO 2101
      DP1=PGSO(KK,1)*DXO(L,K)
      DP2=PGSO(KK,2)*DYO(L,K)
      DP3=PGSO(KK,3)*DZO(L,K)
      PSF(K,L,1,1)=-DP2-DP3
      PSF(K,L,2,2)=-DP1-DP3
      PSF(K,L,3,3)=-DP1-DP2
      PSF(K,L,1,2)=PGSO(KK,2)*DXO(L,K)
      PSF(K,L,1,3)=PGSO(KK,3)*DXO(L,K)
      PSF(K,L,2,1)=PGSO(KK,1)*DYO(L,K)
      PSF(K,L,2,3)=PGSO(KK,3)*DYO(L,K)
      PSF(K,L,3,1)=PGSO(KK,1)*DZO(L,K)
      PSF(K,L,3,2)=PGSO(KK,2)*DZO(L,K)
2101  CONTINUE
      IF(LL.EQ.0) GO TO 210
      IF(K.EQ.L) GO TO 2102
      PD1=PGSO(LL,1)*DXO(K,L)
      PD2=PGSO(LL,2)*DYO(K,L)
      PD3=PGSO(LL,3)*DZO(K,L)
      PSF(K,L,1,1)=PSF(K,L,1,1)-PD2-PD3
      PSF(K,L,2,2)=PSF(K,L,2,2)-PD1-PD3
      PSF(K,L,3,3)=PSF(K,L,3,3)-PD1-PD2
      PSF(K,L,1,2)=PSF(K,L,1,2)+DYO(K,L)*PGSO(LL,1)
      PSF(K,L,1,3)=PSF(K,L,1,3)+DZO(K,L)*PGSO(LL,1)
      PSF(K,L,2,1)=PSF(K,L,2,1)+DXO(K,L)*PGSO(LL,2)
      PSF(K,L,2,3)=PSF(K,L,2,3)+DZO(K,L)*PGSO(LL,2)
      PSF(K,L,3,1)=PSF(K,L,3,1)+DXO(K,L)*PGSO(LL,3)
      PSF(K,L,3,2)=PSF(K,L,3,2)+DYO(K,L)*PGSO(LL,3)
      GO TO 210
2102  CONTINUE
      DO 214 I=1,3
      DO 214 J=1,3
214   AB(I,J)=PSF(K,L,I,J)
      DO 215 I=1,3
      DO 215 J=1,3
215   PSF(K,L,I,J)=AB(I,J)+AB(J,I)
210   CONTINUE
      DO 2151 K=1,NB
      DO 2151 L=1,NB
      IF(K.LE.L) GO TO 2151
      DO 2141 I=1,3
      DO 2141 J=1,3
2141  PSF(K,L,I,J)=PSF(L,K,J,I)
2151  CONTINUE
      DO 3004 K=1,NB
      KK=FI(K)
      DO 3004 L=1,NB
      LL=FI(L)
      IF((KK.EQ.0).AND.(LL.EQ.0)) GO TO 3004
      DO 3003 I=1,3
      DO 3003 J=1,3
      AOO(I,J)=AOO(I,J)-PSF(K,L,I,J)
3003  CONTINUE
3004  CONTINUE
C
C     AOK VECTOR ELEMENT COMPUTATION (3X1)
C
C     AKM SCALAR ELEMENT COMPUTATION
C
      DO 14 M=1,NM
      IQ=M(M)+1
```

```
      AV(M,1)=0.
      AV(M,2)=0.
      AV(M,3)=0.
      DO 7 J=1,NB
      DO 7 I=IQ,NB
      DO 11 N=1,3
      IF(EPS(M,I).EQ.0) GO TO 7
      PSG(J,I,N)=0.
      DO 10 L=1,3
10    PSG(J,I,N)=PSG(J,I,N)+(PS(J,I,N,L)-PSF( J,I,N,L))*GO(M,L)
11    AV(M,N)=AV(M,N)+PSG(J,I,N)
7     CONTINUE
      DO 14 K=1,NH
      IF(K.GT.M) GO TO 14
      JQ=H(K)+1
      AIS(1)=0.
      AIS(2)=0.
      AIS(3)=0.
      DO 15 J=JQ,NB
      DO 15 I=IQ,NB
      IF((EPS(K,J).EQ.0).OR.(EPS(M,I).EQ.0)) GO TO 15
      DO 18 N=1,3
18    AIS(N)=AIS(N)+PSG(J,I,N)
15    CONTINUE
      AS(K,M)=GO(K,1)*AIS(1)+GO(K,2)*AIS(2)+GO(K,3)*AIS(3)
14    CONTINUE
C
C     AOFI AND AOFR MATRIX COMPUTATION (3XNKT)
C
      DO 219 K=1,NF
      JK=F(K,3)
      JQ=F(K,1)+1
      DO 222 I=1,3
      DO 222 J=1,3
222   AB(I,J)=0.
      DO 221 L=1,NB
      AB(1,2)=AB(1,2)-DZO(L,JQ)
      AB(1,3)=AB(1,3)+DYO(L,JQ)
221   AB(2,3)=AB(2,3)-DXO(L,JQ)
      AB(2,1)=-AB(1,2)
      AB(3,1)=-AB(1,3)
      AB(3,2)=-AB(2,3)
      DO 220 I=1,3
      DO 220 J=1,JK
      AOFR(K,I,J)=DLKRO(K,I,J)
      AOFI(K,I,J)=DLKIO(K,I,J)
      DO 220 L=1,3
      AOFR(K,I,J)=AOFR(K,I,J) - AB(I,L)*GKOS(K,L,J)
220   AOFI(K,I,J)=AOFI(K,I,J) - AB(I,L)*PKOS(K,L,J)
219   CONTINUE
C
C     AKFR VECTOR COMPUTATION (1XNKT) (FLEX,COUPLING WITH RIGID SUBSTRUCTURE
C
C     AKFI VECTOR COMPUTATION (1XNKT) (FLEX,COUPLING WITH RIGID SUBSTRUCTURE
C
      DO 224 K=1,NF
      JK=F(K,3)
      JQ=F(K,1)+1
      DO 2245 J=1,JK
      ZSR(K,J)=0.
2245  ZSI(K,J)=0.
      DO 224 M=1,NM
      DO 231 I=1,3
      DO 231 J=1,3
231   AB(I,J)=0.
      DO 226 L=1,NB
      IF(EPS(M,L).EQ.0) GO TO 226
      AB(1,2)=AB(1,2)-DZO(L,JQ)
      AB(1,3)=AB(1,3)+DYO(L,JQ)
```

```
      AB(2,3)=AB(2,3)-DXO(L,JQ)
226   CONTINUE
      AB(2,1)=-AB(1,2)
      AB(3,1)=-AB(1,3)
      AB(3,2)=-AB(2,3)
      DO 228 I=1,3
      DO 228 J=1,JK
      DUR(I,J)=DLKRO(K,I,J)
      DUI(I,J)=DLKIO(K,I,J)
      IF(EPS(M,K).EQ.0) DUR(I,J)=0.
      IF(EPS(M,K).EQ.0) DUI(I,J)=0.
      DO 228 L=1,3
      DUR(I,J)=DUR(I,J)-AB(I,L)*GKOS(K,L,J)
228   DUI(I,J)=DUI(I,J)-AB(I,L)*PKOS(K,L,J)
      DO 2241 J=1,JK
      DO 2241 I=1,3
      ZSR(K,J)=ZSR(K,J)+DUR(I,J)*WGJ(M,I)
2241  ZSI(K,J)=ZSI(K,J)+DUI(I,J)*WGJ(M,I)
      DO 229 J=1,JK
      AKFR(K,M,J)=0.
      AKFI(K,M,J)=0.
      DO 229 I=1,3
      AKFR(K,M,J)=AKFR(K,M,J)+GO(M,I)*DUR(I,J)
229   AKFI(K,M,J)=AKFI(K,M,J)+GO(M,I)*DUI(I,J)
224   CONTINUE
C
C     COMPUTE CORRECTION ELEMENTS FOR (E) VECTOR
C
      DO 41 J=2,NB
      JK=HI(J)
      DO 411 M=1,3
411   CW(J,M)=0.
      DO 42 K=1,JK
      IF(EPS(K,J).EQ.0) GO TO 42
      CW(J,1)=CW(J,1)+WGJ(K,1)
      CW(J,2)=CW(J,2)+WGJ(K,2)
      CW(J,3)=CW(J,3)+WGJ(K,3)
42    CONTINUE
41    CONTINUE
      DO 40 I=1,NB
      EA(1)=0.
      EA(2)=0.
      EA(3)=0.
      DO 401 J=2,NB
      DO 4507 M=1,3
      DO 4507 L=1,3
4507  EA(M)=EA(M)+(PS(I,J,M,L)-PSF( I,J,M,L))*CW(J,L)
401   CONTINUE
      K1=3*(I-1)
      E(K1+1,1)=E(K1+1,1)-EA(1)
      E(K1+2,1)=E(K1+2,1)-EA(2)
      E(K1+3,1)=E(K1+3,1)-EA(3)
40    CONTINUE
      DO 55 MI=1,3
55    EC(MI)=E(MI,1)
      DO 52 J=2,NB
      DO 52 M=1,3
      K1=3*(J-1)+M
52    EC(M)=EC(M)+E(K1,1)
      I=0
      DO 60 K=1,NH
      JK=H(K)+1
      IF(PI(K).NE.0) GO TO 60
      I=I+1
      EC(I+3)=0.
      DO 601 M=1,3
601   CE(M)=0.
      DO 61 J=JK,NB
      IF(EPS(K,J).EQ.0) GO TO 61
```

```
      DO 65 M=1,3
      J1=3*(J-1)+M
   65 CE(M)=CE(M)+E(J1,I)
   61 CONTINUE
      DO 66 L=1,3
   66 EC(I+3)=EC(I+3)+GO(K,L)*CE(L)
      EC(I+3)=EC(I+3)+TH(K)
   60 CONTINUE
      DO 610 I=1,3
      DO 610 J=1,NH
      IF(PI(J).EQ.0) GO TO 610
      EC(I)=EC(I)-AV(J,I)*GMDD(J)
  610 CONTINUE
      K=0
      IV=3
      DO 612 I=1,NH
      IF(PI(I).NE.0) GO TO 612
      K=K+1
      IV=IV+1
      DO 611 J=1,NH
      IF(PI(J).EQ.0) GO TO 611
      IF(I.GT.J) AS(I,J)=AS(J,I)
      EC(K+3)=EC(K+3)-AS(I,J)*GMDD(J)
  611 CONTINUE
  612 CONTINUE
C
C     COMPUTE RT. HAND SIDE OF APPENDAGE EQUATIONS (IN APPEND. COORDS.)
C
      DO 477 K=1,NF
      DO 479 I=1,3
  479 CDU(K,I)=0.
      DO 478 L=1,NF
      IF(K.EQ.L) GO TO 478
      CDU(K,2)=CDU(K,2)+DUYO(L)
      CDU(K,1)=CDU(K,1)+DUXO(L)
      CDU(K,3)=CDU(K,3)+DUZO(L)
  478 CONTINUE
  477 CONTINUE
      DO 483 K=1,NF
      I=F(K,1)+1
      M=HI(I)
      CQ(1)=(FTXO+CDU(K,1))/TM + CWWD(I,1)
      CQ(2)=(FTYO+CDU(K,2))/TM + CWWD(I,2)
      CQ(3)=(FTZO+CDU(K,3))/TM + CWWD(I,3)
      IF(I.EQ.1) GO TO 4840
      DO 484 J=1,3
      VE(K,J)=-WWDE(K,J)
      DO 484 L=1,3
  484 VE(K,J)=VE(K,J)+T(M,J,L)*CQ(L)
      GO TO 483
 4840 CONTINUE
      DO 4841 J=1,3
 4841 VE(K,J)=CQ(J)-WWDE(K,J)
  483 CONTINUE
      DO 485 K=1,NF
      NL=F(K,2)
      I=F(K,1)+1
      M=HI(I)
      R1=SR(K,1)
      R2=SR(K,2)
      R3=SR(K,3)
      IF(I.EQ.1) GO TO 4870
      DO 487 J=1,3
  487 WW(J)=T(M,J,1)*WXO(I)+T(M,J,2)*WYO(I)+T(M,J,3)*WZO(I)
      GO TO 4872
 4870 CONTINUE
      WW(1)=WXO(I)
      WW(2)=WYO(I)
      WW(3)=WZO(I)
 4872 CONTINUE
```

```
      W11=WW(1)**2-R1**2
      W22=WW(2)**2-R2**2
      W33=WW(3)**2-R3**2
      W12=WW(1)*WW(2)-R1*R2
      W13=WW(1)*WW(3)-R1*R3
      W23=WW(2)*WW(3)-R2*R3
      DO 486 N=1,NL
      N6=6*(N-1)
      DO 488 J=1,3
      JN=N6+J
      JM=JN+3
      VB(K,JN)=FF(K,N,J)
488   VB(K,JM)=TF(K,N,J)
      VB(K,N6+1)=VB(K,N6+1)-MF(K,N,7)*(-RF(K,N,1)*(W33+W22)+RF(K,N,2)*W1
     $2+RF(K,N,3)*W13)
      VB(K,N6+2)=VB(K,N6+2)-MF(K,N,7)*(-RF(K,N,2)*(W33+W11)+RF(K,N,1)*W1
     $2+RF(K,N,3)*W23)
      VB(K,N6+3)=VB(K,N6+3)-MF(K,N,7)*(-RF(K,N,3)*(W11+W22)+RF(K,N,1)*W1
     $3+RF(K,N,2)*W23)
      CE(1)= MF(K,N,1)*WW(1)-MF(K,N,4)*WW(2)-MF(K,N,5)*WW(3)
      CE(2)=-MF(K,N,4)*WW(1)+MF(K,N,2)*WW(2)-MF(K,N,6)*WW(3)
      CE(3)=-MF(K,N,5)*WW(1)-MF(K,N,6)*WW(2)+MF(K,N,3)*WW(3)
      CL(1)=MF(K,N,1)*R1-MF(K,N,4)*R2-MF(K,N,5)*R3
      CL(2)=-MF(K,N,4)*R1+MF(K,N,2)*R2-MF(K,N,6)*R3
      CL(3)=-MF(K,N,5)*R1-MF(K,N,6)*R2+MF(K,N,3)*R3
      VB(K,N6+4)=VB(K,N6+4)-(WW(2)*CE(3)-WW(3)*CE(2))
     $+(R2*CL(3)-R3*CL(2))
      VB(K,N6+5)=VB(K,N6+5)-(WW(3)*CE(1)-WW(1)*CE(3))
     $+(R3*CL(1)-R1*CL(3))
      VB(K,N6+6)=VB(K,N6+6)-(WW(1)*CE(2)-WW(2)*CE(1))
     $+(R1*CL(2)-R2*CL(1))
486   CONTINUE
485   CONTINUE
      NV=IV
      DO 491 K=1,NF
      JN=F(K,3)
      NL=F(K,2)
      NL6=6*NL
      DO 492 J=1,JN
      IL=NV+J
      IO=IL+NTMO
      VV1=(ET(K,J)+ZF(K,J)*DT(K,J))*2.
      VV2=(DT(K,J)-ZF(K,J)*ET(K,J))*2.
      DO 493 N=1,NL6
      VV1=VV1+EI(K,N,J)*VB(K,N)*2.
493   VV2=VV2-ER(K,N,J)*VB(K,N)*2.
      DO 494 N=1,3
      VV1=VV1-GK(K,N,J)*VE(K,N)
494   VV2=VV2+PK(K,N,J)*VE(K,N)
      VV1=-WF(K,J)*VV1-ZSR(K,J)
      VV2=+WF(K,J)*VV2-ZSI(K,J)
      EC(IL)=VV1
      EC(IO)=VV2
      DO 4920 L=1,NH
      IF(PI(L).EQ.0) GO TO 4920
      EC(IL)=EC(IL)-AKFR(L,K,J)*GMDD(L)
      EC(IO)=EC(IO)-AKFI(L,K,J)*GMDD(L)
4920  CONTINUE
492   CONTINUE
491   NV=NV+JN
C
C     ENTER CONSTANTS INTO FLEX. BODY PORTION OF COEFF. MATRIX A
C
      NV=IV
      DO 462 K=1,NF
      NL=F(K,3)
      DO 463 I=1,NL
      IL=NV+I
      IO=IL+NTMO
```

```
      DO 463 J=1,NL
      JL=NV+J
      JO=JL+NTMO
      A(IL,JL)=0.
      A(IL,JO)=0.
      A(IO,JL)=0.
      A(IO,JO)=0.
      IF(I.EQ.J) A(IL,JL)=2.
      IF(I.EQ.J) A(IO,JO)=2.
463   CONTINUE
462   NV=NV+NL
C
C     ENTER COEFF. WHICH COUPLE REF. BODY AND FLEX. APPENDAGES INTO A
C
      NV=IV
      DO 464 K=1,NF
      NL=F(K,3)
      DO 465 J=1,3
      DO 465 I=1,NL
      IL=NV+I
      IO=IL+NTMO
      A(IL,J)=AOFR(K,J,I)
      A(J,IL)=A(IL,J)
      A(IO,J)=AOFI(K,J,I)
465   A(J,IO)=A(IO,J)
464   NV=NV+NL
C
C     ENTER COEFF. WHICH COUPLE SUBSTR. BODIES AND FLEX. APPEND. INTO A
C
      NV=IV
      DO 466 K=1,NF
      NL=F(K,3)
      JI=0
      DO 467 J=1,NH
      IF(PI(J).NE.0) GO TO 467
      JI=JI+1
      DO 4671 I=1,NL
      IL=NV+I
      IO=IL+NTMO
      A(IL,JI+3)=AKFR(K,J,I)
      A(IO,JI+3)=AKFI(K,J,I)
      A(JI+3,IL)=A(IL,JI+3)
      A(JI+3,IO)=A(IO,JI+3)
4671  CONTINUE
467   CONTINUE
466   NV=NV+NL
C
C     CALCULATE FLEX. BODY COUPLING COEFF. AND ENTER INTO A MATRIX
C
      NCO=IV
      DO 473 L=1,NF
      NL=F(L,3)
      NRO=IV
      DO 474 K=1,NF
      NR=F(K,3)
      IF(K.EQ.L) GO TO 474
      DO 475 I=1,NR
      IK=NRO+I
      IO=IK+NTMO
      DO 475 J=1,NL
      JK=NCO+J
      JO=JK+NTMO
      A(IK,JK)=0.
      A(IO,JK)=0.
      A(IK,JO)=0.
      A(IO,JO)=0.
      DO 4750 N=1,3
      A(IK,JK)=A(IK,JK)-GKOS(K,N,I)*GKOS(L,N,J)/TM
      A(IO,JK)=A(IO,JK)-PKOS(K,N,I)*GKOS(L,N,J)/TM
```

ORIGINAL PAGE IS OF POOR QUALITY

```
      A(IK,JO)=A(IK,JO)-GKOS(K,N,I)*PKOS(L,N,J)/TM
      A(IO,JO)=A(IO,JO)-PKOS(K,N,I)*PKOS(L,N,J)/TM
4750  CONTINUE
      A(JK,IK)=A(IK,JK)
      A(JK,IO)=A(IO,JK)
      A(JO,IK)=A(IK,JO)
      A(JO,IO)=A(IO,JO)
475   CONTINUE
474   NRO=NRO+NR
473   NCO=NCO+NL
C
C     LOAD SYSTEM MATRIX (A) WITH AOO,AOK,AKM ELEMENTS
C
      DO 23 I=1,3
      DO 23 J=1,3
23    A(I,J)=AOO(I,J)
      DO 24 I=1,3
      K=0
      DO 24 J=1,NH
      IF(PI(J).NE.0) GO TO 24
      K=K+1
      A(K+3,I)=AV(J,I)
      A(I,K+3)=AV(J,I)
24    CONTINUE
      K=0
      DO 250 I=1,NH
      IF(PI(I).NE.0) GO TO 250
      K=K+1
      L=0
      DO 25 J=1,NH
      IF(PI(J).NE.0) GO TO 25
      L=L+1
      IF(K.GT.L) GO TO 26
      A(K+3,L+3)=AS(I,J)
      GO TO 25
26    A(K+3,L+3)=A(L+3,K+3)
25    CONTINUE
250   CONTINUE
C
C     ANGULAR MOMENTUM OF THE SYSTEM
C
      IF(PI(NH+1).NE.1) GO TO 8752
      DO 5651 I=1,3
      HH(I)=0.
      DO 5651 J=1,3
5651  HH(I)=HH(I)+A(I,J)*WO(J)
      DO 5652 I=1,3
      DO 5652 J=1,QH
5652  HH(I)=HH(I)+AV(J,I)*GMD(J)
      DO 5653 I=1,3
      DO 5653 K=1,NF
      NL=F(K,3)
      DO 5654 J=1,NL
5654  HH(I)=HH(I)+AOFR(K,I,J)*DT(K,J)+AOFI(K,I,J)*ET(K,J)
5653  CONTINUE
      HM=SQRT(HH(1)**2 + HH(2)**2 + HH(3)**2)
8752  CONTINUE
C
C     SOLVE SYSTEM MATRIX FOR REFERENCE BODY ANG. ACCELERATION AND HINGE
C                  (RELATIVE) ROTATIONAL ACCELERATIONS
C
      NT=V+NT2
      IT=IV+NT2
      KV=IV
      CALL CHOLD(592,A,ST,IT,EC,D.,1.D-7)
      DO 910 J=NT,4,-1
      IF(J.LE.V) GO TO 913
      JV=J-(V-IV)
      EC(J)=EC(JV)
```

```
      GO TO 910
913   CONTINUE
      K=J-3
      IF(PI(K).NE.0) GO TO 911
      EC(J)=EC(KV)
      KV=KV-1
      GO TO 910
911   EC(J)=GMOD(K)
910   CONTINUE
      DO 9003 I=1,V
9003  ROOT(I)=EC(I)
      I=V
      DO 9001 K=1,NF
      NL=F(K,3)
      DO 9002 N=1,NL
      IO=I+N
      IL=IO+NTMO
      DTD(K,N)=EC(IO)
9002  ETD(K,N)=EC(IL)
9001  I=I+NL
92    CONTINUE
      RETURN
      END
```

DIAGNOSTICS

ATION TIME = 44.48 SUPS

CSSL.TRAN.CSSL

Appendix D

Subroutine MBDYFN Listing and User Requirements

Subroutine Entry Statements

CALL MBDYFN(NC, H, MB, MS, PB, PS, G, PI,
NF, F, EIG, REC, RF, WF, ZF)

CALL MRATE(NC, TH, TB, TS, FB, FS, TF, FF, GM, GMD,
GMDD, ET, ETD, WO, WDOT, ETD, HM)

Input / Output Variable Type and Storage Specifications

INTEGER NC, NF, H(n_c, 2), F(n_f, 3), PI(n + 1)

REAL MB(7), MS(n_c, 7), PB(n_c, 3), PS(n_c, n_c, 3),
G(n, 3), TH(n), TB(3), TS(n_c, 3), FB(3), FS(n_c, 3),
GM(n), GMD(n), GMDD(n), EIG(n_f, 6n_k, N_k),
REC(n_f, 6, N_k), RF(n_f, n_k, 3), WF(n_f, N_k),
ZF(n_f, N_k), TF(n_f, n_k, 3), FF(n_f, n_k, 3),
ET(n_f, N_k), ETD(n_f, N_k), WO(3).

DOUBLE PRECISION WDOT(n + 3), ETDD(n_f, N_k)

External Subroutines Called

CHOLD—(see Appendix C and statement 1013)

Subroutine Setup

Insert the Fortran statement

PARAMETER QC = n_c, QH = n, QF = n_f, NK = n_k, NKT = N_k

(If more than one appendage is present, use the *largest* n_k and N_k for the PARAMETER statement to provide sufficient storage.)

Data Restrictions

$n > 1, n_f > 1, n_c > 1, n_k > 1, N_k > 1$

Core Storage Required

Code: 4500 words

Data: ~500 words (minimum; increases with n, n_f).

Listing

```
      SUBROUTINE MBDYFN(NC,C,MB,MA,PB,PA,G,PI,NF,F,EIG,REC,RF,WF,ZF)
C
C     ADJUSTABLE DIMENSIONS
C
      INTEGER PI(1),C(NC,2)
      REAL MB(1),MA(NC,7),PB(NC,3),PA(NC,NC,3)
      PARAMETER QC=1,QH=2,QF=1,NK=1,NKT=7
      PARAMETER N6K=6*NK,S=QC+1,V=QH+3,V4=4*V,S3=3*S,Q=QH,NH=QH
      PARAMETER ST=V+QF*NKT,S4=4*ST
C
C     ADDITIONAL DIMENSIONED VARIABLES
C
      DOUBLE PRECISION A(ST,ST),BMASS(S)
      INTEGER EPS(Q,S),CPS(QC,S),H(Q),HI(S),FI(S),F(NF,3)
      REAL ADD(3,3),AB(3,3),AOF(QF,3,NKT),AKF(QF,QH,NKT),AS(Q,Q),AV(Q,3)
     S,AIS(3),CE(3),CK(QF,3),CW(3),CWWD(S,3),CW(S,3),DX(S,S),DY(S,S),DZ(
     SS,S),DXO(S,S),DYO(S,S),DZO(S,S),DLK(WF,3,NKT),DLKO(WF,3,NKT),DUR(3
     S,NKT),EA(3),EIG(NF,N6K,NKT),FEXO(S),FEYO(S),FEZO(S),FS(S,3),GU(Q,3
     S),GG(Q,3),G(Q,3),IXX(S),IYY(S),IZZ(S),IXY(S),IXZ(S),IYZ(S),LX(S,S)
     S,LY(S,S),LZ(S,S),MSB(S),MCK(WF,3),PH(S,3,3),PSG(S,S,3),PS(S,S,3,3)
     S,PK(QF,3,NKT),PGSO(QF,3),PSF(S,S,3,3),PKO(QF,3,NKT),RF(NF,NK,3),RE
     SC(NF,6,NKT),TXO(S),TYO(S),TZO(S),T(Q,3,3),TS(S,3),U(WF,NK,3),VE(QF
     S,3),VB(QF,N6K),WF(NF,NKT),WGJ(QH,3),ZF(NF,NKT),ZSR(WF,NKT),HH(3)
      EQUIVALENCE (A,PS),(LX,DXO),(LY,DYO),(LZ,DZO)
      NB=NC+1
C
C     DEFINE EPS(K,J) USING C
C
      DO 86 K=1,NC
      DO 86 J=2,NB
      IF(K.EQ.(J-1)) CPS(K,J)=1
      IF(K.LT.(J-1)) GO TO 87
      GO TO 86
   87 CONTINUE
      JO=K+1
      JI=J-1
      DO 89 L=JO,JI
      IF(K.GT.(L-1)) GO TO 89
      IF((CPS(K,L).EQ.1).AND.(C(J-1,1).EQ.(L-1))) CPS(K,J)=1

   89 CONTINUE
   86 CONTINUE
      L=0
      DO 1 J=1,NC
      KK=C(J,2)
      DO 1 K=1,KK
      L=L+1
      DO 1 I=1,NB
    1 EPS(L,I)=CPS(J,I)
C
C     COMPUTE H(I)=C, WHERE I=HINGE LABEL AND C=CONNECTION LABEL
C
      I=0
      DO 8 J=2,NB
      KK=C(J-1,2)
      DO 8 K=1,KK
      I=I+1
    8 H(I)=J-1
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C
C       COMPUTE HI(I)=J, WHERE I=BODY LABEL+1 AND J=NEAREST HINGE LABEL
C
        HI(1)=1
        HI(NB)=NH
        DO 47 I=NH,1
        IF(I.EQ.1) GO TO 47
        K1=H(I)
        K2=H(I-1)
        IF(K1.EQ.K2) GO TO 47
        HI(K2+1)=I-1
47      CONTINUE
C
C       DEFINE FI(J)=K, WHERE J=BODY-LABEL+1 AND K IS APPENDAGE-LABEL
C                  (IF K=0, BODY HAS NO FLEX. APPENDAGE)
C
        DO 239 N=1,NB
239     FI(N)=0
        DO 242 K=1,NF
        JN=F(K,1)+1
242     FI(JN)=K
        NF=NF
        NB=NB
C
C       DEFINE SUBSTRUCTURE MASSES
C
        MSB(1)=MB(7)
        DO 248 N=2,NB
248     MSB(N)=MA(N-1,7)
C
C       TOTAL NUMBER OF FLEX. APPENDAGE MODES TO BE RETAINED
C
        NTMO=0
        DO 461 K=1,NF
461     NTMO=NTMO+F(K,3)
C
C       INITIAL CALCULATION OF BARYCENTER VECTORS W.R.T. BODY C.G.S
C                             AND HINGE POINTS
C
        IXX(1)=MB(1)
        IYY(1)=MB(2)
        IZZ(1)=MB(3)
        IXY(1)=MB(4)
        IXZ(1)=MB(5)
        IYZ(1)=MB(6)
        BMASS(1)=MB(7)
        TM=BMASS(1)
        DO 35 J=2,NB
        IXX(J)=MA(J-1,1)
        IYY(J)=MA(J-1,2)
        IZZ(J)=MA(J-1,3)
        IXY(J)=MA(J-1,4)
        IXZ(J)=MA(J-1,5)
        IYZ(J)=MA(J-1,6)
        BMASS(J)=MA(J-1,7)
35      TM=TM+BMASS(J)
        DO 149 I=1,NB
        I1=I-1
        DO 149 J=1,NB
        J1=J-1
        IF(I.EQ.J) GO TO 163
        IF(I.GT.J) GO TO 70
        IF(I.EQ.1) GO TO 80
        IF(CPS(I1,J).EQ.1) GO TO 400
70      LX(I,J)=PA(I1,I1,1)
        LY(I,J)=PA(I1,I1,2)
        LZ(I,J)=PA(I1,I1,3)
        GO TO 149
400     CONTINUE
```

```
      DO 600 K=1,J1
      IF(CPS(K,J).EQ.1) GO TO 500
600   CONTINUE
      GO TO 149
500   LX(I,J)=PA(I1,K,1)
      LY(I,J)=PA(I1,K,2)
      LZ(I,J)=PA(I1,K,3)
      GO TO 149
80    DO 90 L=1,J1
      IF(CPS(L,J).EQ.1) GO TO 101
90    CONTINUE
      GO TO 149
101   LX(I,J)=PB(L,1)
      LY(I,J)=PB(L,2)
      LZ(I,J)=PB(L,3)
      GO TO 149
103   LX(I,J)=0.
      LY(I,J)=0.
      LZ(I,J)=0.
149   CONTINUE
      DO 13 N=1,NB
      DO 13 J=1,NB
      DX(N,J)=LX(N,J)
      DY(N,J)=LY(N,J)
      DZ(N,J)=LZ(N,J)
      DO 13 K=1,NB
      DX(N,J)=DX(N,J)-(BMASS(K)/TM)*LX(N,K)
      DY(N,J)=DY(N,J)-(BMASS(K)/TM)*LY(N,K)
13    DZ(N,J)=DZ(N,J)-(BMASS(K)/TM)*LZ(N,K)
C
C     CALCULATION OF AUGMENTED INERTIA DYADICS FOR EACH BODY
C
      DO 31 N=1,NB
      PH(N,1,1)=IXX(N)
      PH(N,1,2)=-IXY(N)
      PH(N,1,3)=-IXZ(N)
      PH(N,2,2)=IYY(N)
      PH(N,2,3)=-IYZ(N)
      PH(N,3,3)=IZZ(N)
      DO 30 J=1,NB
      PH(N,1,1)=PH(N,1,1)+BMASS(J)*(DY(N,J)**2+DZ(N,J)**2)
      PH(N,1,2)=PH(N,1,2)-BMASS(J)*DX(N,J)*DY(N,J)
      PH(N,1,3)=PH(N,1,3)-BMASS(J)*DX(N,J)*DZ(N,J)
      PH(N,2,2)=PH(N,2,2)+BMASS(J)*(DX(N,J)**2+DZ(N,J)**2)
      PH(N,2,3)=PH(N,2,3)-BMASS(J)*DY(N,J)*DZ(N,J)
30    PH(N,3,3)=PH(N,3,3)+BMASS(J)*(DX(N,J)**2+DY(N,J)**2)
      PH(N,2,1)=PH(N,1,2)
      PH(N,3,1)=PH(N,1,3)
31    PH(N,3,2)=PH(N,2,3)
C
C     DEFINE PK(3 X NKT ARRAY)
C     DEFINE DLK=TRANSPOSE MATRIX (3 X NKT ARRAY)
C
      DO 201 K=1,NF
      JNT=F(K,3)
      DO 201 I=1,3
      DO 201 J=1,JNT
      PK(K,I,J)=REC(K,I,J)
201   DLK(K,I,J)=REC(K,I+3,J)
      RETURN
      ENTRY MRATE(NC,TH,TB,TA,FB,FA,TF,FF,GM,GMD,GMDD,ET,ETD,WO,WDOT,ETD
     $D,HM)
      REAL TF(QF,NK,3),FF(QF,NK,3),ET(QF,NKT),ETD(QF,NKT),TB(3),TA(NC,3)
     $,FB(3),FA(NC,3),GM(1),GMD(1),GMDD(1),TH(1),WO(3),WXO(S),WYO(S),WZO
     $(S),E(S3,1)
      DOUBLE PRECISION EC(ST),ETDD(QF,NKT),WDOT(V)
C
C     BODY-TO-BODY COORDINATE TRANSFORMATION MATRICES
C
```

```
      DO 335 J=1,NH
      MM=J-1
      N=H(J)+1
      SGM=SIN(GM(J))
      CGM=COS(GM(J))
      CGM1=1.-CGM
      G1=CGM1*G(J,1)
      G2=CGM1*G(J,2)
      G3=CGM1*G(J,3)
      SG1=SGM*G(J,1)
      SG2=SGM*G(J,2)
      SG3=SGM*G(J,3)
      G1S=G1*G(J,1)
      G2S=G2*G(J,2)
      G3S=G3*G(J,3)
      G12=G1*G(J,2)
      G13=G1*G(J,3)
      G23=G2*G(J,3)
      AB(1,1)=CGM+G1S
      AB(1,2)=SG3+G12
      AB(1,3)=-SG2+G13
      AB(2,1)=-SG3+G12
      AB(2,2)=CGM+G2S
      AB(2,3)=SG1+G23
      AB(3,1)=SG2+G13
      AB(3,2)=-SG1+G23
      AB(3,3)=CGM+G3S
      IF(J.EQ.1) GO TO 3350
      DO 321 L=MM,1
      IF(EPS(L,N).EQ.1) GO TO 322
  321 CONTINUE
      GO TO 3350
  322 K=L
      DO 334 L=1,3
      DO 334 M=1,3
      T(J,L,M)=0.
      DO 334 I=1,3
  334 T(J,L,M)=T(J,L,M)+AB(L,I)*T(K,I,M)
      GO TO 335
 3350 CONTINUE
      DO 3351 L=1,3
      DO 3351 M=1,3
 3351 T(J,L,M)=AB(L,M)
  335 CONTINUE
C
C     COORD. TRANSFORMATION OF G VECTORS (TO REF. BODY FRAME)
C
      DO 362 I=1,NH
      DO 362 J=1,3
      GO(I,J)=0.
      DO 362 K=1,3
      GO(I,J)=GO(I,J)+T(I,K,J)*G(I,K)
  362 CONTINUE
C
C     ANG. VELOCITY COMPONENTS OF EACH BODY (IN REF. BODY FRAME)
C
      DO 366 K=1,NH
      GG(K,1)=GMD(K)*GO(K,1)
      GG(K,2)=GMD(K)*GO(K,2)
  366 GG(K,3)=GMD(K)*GO(K,3)
      DO 361 J=1,NB
      KV=HI(J)
      WXO(J)=WO(1)
      WYO(J)=WO(2)
      WZO(J)=WO(3)
      DO 36 K=1,KV
      IF(EPS(K,J).EQ.0) GO TO 36
      WXO(J)=WXO(J)+GG(K,1)
      WYO(J)=WYO(J)+GG(K,2)
```

```
      WZO(J)=WZO(J)+GG(K,3)
36    CONTINUE
361   CONTINUE
C
C     ANG. VELOCITY COMPONENTS AT EACH HINGE (IN REF. BODY FRAME)
C
      DO 3666 M=1,NH
      M1=M+1
      MC=H(M)+1
      NI=HI(MC)
      WHXO=WXO(MC)
      WHYO=WYO(MC)
      WHZO=WZO(MC)
      IF(NI.EQ.M) GO TO 3667
      DO 3668 N=M1,NI
      WHXO=WHXO-GG(N,1)
      WHYO=WHYO-GG(N,2)
3668  WHZO=WHZO-GG(N,3)
3667  CONTINUE
      WGJ(M,1)=GG(M,3)*WHYO-GG(M,2)*WHZO
      WGJ(M,2)=GG(M,1)*WHZO-GG(M,3)*WHXO
      WGJ(M,3)=GG(M,2)*WHXO-GG(M,1)*WHYO
3666  CONTINUE
C
C     TRANSFORM PK AND DLK TO REF. BODY BASIS
C
      DO 468 K=1,NF
      KK=F(K,1)+1
      JNT=F(K,3)
      IF(KK.EQ.1) GO TO 4720
      M=HI(KK)
      DO 469 I=1,3
      DO 469 J=1,JNT
      DLKO(K,I,J)=0.
      PKO(K,I,J)=0.
      DO 469 L=1,3
      DLKO(K,I,J)=DLKO(K,I,J)+T(M,L,I)*DLK(K,L,J)
469   PKO(K,I,J)=PKO(K,I,J)+T(M,L,I)*PK(K,L,J)
      GO TO 468
4720  CONTINUE
      DO 4721 I=1,3
      DO 4721 J=1,JNT
      DLKO(K,I,J)=DLK(K,I,J)
4721  PKO(K,I,J)=PK(K,I,J)
468   CONTINUE
C
C     COMPUTE TOTAL EXTERNAL FORCE ON EACH SUBSTRUCTURE (IN REF. COORD.)
C
      FEXO(1)=FB(1)
      FEYO(1)=FB(2)
      FEZO(1)=FB(3)
      IF(FI(1).EQ.0) GO TO 254
      IL=FI(1)
      JN=F(IL,2)
      DO 253 J=1,JN
      FEXO(1)=FEXO(1)+FF(IL,J,1)
      FEYO(1)=FEYO(1)+FF(IL,J,2)
253   FEZO(1)=FEZO(1)+FF(IL,J,3)
254   CONTINUE
      FS(1,1)=FEXO(1)
      FS(1,2)=FEYO(1)
      FS(1,3)=FEZO(1)
      DO 246 N=2,NB
      K=N-1
      DO 2460 L=1,3
2460  FS(N,L)=FA(K,L)
      IF(FI(N).EQ.0) GO TO 246
      IL=FI(N)
      JN=F(IL,2)
```

```
      DO 245 J=1,JN
      DO 245 I=1,3
245   FS(N,I)=FS(N,I)+FF(IL,J,I)
246   CONTINUE
C
C     COMPUTE TRANSL. AND ROTAT. DISPLACEMENTS OF APPENDAGE SUB-BODIES
C
      DO 232 K=1,NF
      JN=F(K,2)
      LK=F(K,3)
      DO 233 J=1,JN
      DO 233 I=1,3
      U(K,J,I)=0.
      ID=(J-1)*6+I
      DO 233 L=1,LK
233   U(K,J,I)=U(K,J,I)+EIG(K,ID,L)*ET(K,L)
232   CONTINUE
C
C     COMPUTE C.M. PERTURBATION (FROM NOM. UNDEFORMED LOCATION) ON EACH
C              SUBSTRUCTURE WITH AN APPENDAGE (LOCAL COORDS.)
C
      DO 262 K=1,NF
      IK=F(K,1)+1
      JN=F(K,3)
      DO 263 I=1,3
263   MCK(K,I)=0.
      DO 265 J=1,JN
      DO 265 I=1,3
265   MCK(K,I)=MCK(K,I)-PK(K,I,J)*ET(K,J)
      DO 266 I=1,3
266   CK(K,I)=MCK(K,I)/MSB(IK)
262   CONTINUE
C
C     COMPUTE TOTAL EXTERNAL TORQUE ON EACH SUBSTRUCTURE W.R.T. ITS
C              INSTANTANEOUS C.M. (IN LOCAL COORD.)
C
      DO 268 L=1,3
268   TS(1,L)=TB(L)
      DO 267 N=2,NB
      K=N-1
      DO 267 L=1,3
267   TS(N,L)=TA(K,L)
      DO 2670 N=1,NB
      IL=FI(N)
      IF(IL.EQ.0) GO TO 2670
      JN=F(IL,2)
      DO 2671 J=1,JN
      DO 2671 L=1,3
2671  TS(N,L)=TS(N,L)+TF(IL,J,L)
2670  CONTINUE
      DO 269 N=1,NB
      K=FI(N)
      IF(K.EQ.0) GO TO 269
      TS(N,1)=TS(N,1)+CK(K,2)*FS(N,3)-CK(K,3)*FS(N,2)
      TS(N,2)=TS(N,2)+CK(K,3)*FS(N,1)-CK(K,1)*FS(N,3)
      TS(N,3)=TS(N,3)+CK(K,1)*FS(N,2)-CK(K,2)*FS(N,1)
269   CONTINUE
      DO 271 N=1,NB
      K=FI(N)
      IF(K.EQ.0) GO TO 271
      JN=F(K,2)
      DO 272 J=1,JN
      RUX=RF(K,J,1)+U(K,J,1)
      RUY=RF(K,J,2)+U(K,J,2)
      RUZ=RF(K,J,3)+U(K,J,3)
      TS(N,1)=TS(N,1)+RUY*FF(K,J,3)-RUZ*FF(K,J,2)
      TS(N,2)=TS(N,2)+RUZ*FF(K,J,1)-RUX*FF(K,J,3)
272   TS(N,3)=TS(N,3)+RUX*FF(K,J,2)-RUY*FF(K,J,1)
271   CONTINUE
```

```
C
C       TRANSFORM VECTORS TO REF. BODY FRAME
C
        TXO(1)=TS(1,1)
        TYO(1)=TS(1,2)
        TZO(1)=TS(1,3)
        DO 17 I=2,NB
        M=MI(I)
        K=I-1
        L=C(K,I)+1
        FEXO(I)=T(M,1,1)*FS(I,1)+T(M,2,1)*FS(I,2)+T(M,3,1)*FS(I,3)
        FEYO(I)=T(M,1,2)*FS(I,1)+T(M,2,2)*FS(I,2)+T(M,3,2)*FS(I,3)
        FEZO(I)=T(M,1,3)*FS(I,1)+T(M,2,3)*FS(I,2)+T(M,3,3)*FS(I,3)
        TXO(I) =T(M,1,1)*TS(I,1)+T(M,2,1)*TS(I,2)+T(M,3,1)*TS(I,3)
        TYO(I) =T(M,1,2)*TS(I,1)+T(M,2,2)*TS(I,2)+T(M,3,2)*TS(I,3)
        TZO(I) =T(M,1,3)*TS(I,1)+T(M,2,3)*TS(I,2)+T(M,3,3)*TS(I,3)
        DXO(I,I)=T(M,1,1)*DX(I,I)+T(M,2,1)*DY(I,I)+T(M,3,1)*DZ(I,I)
        DYO(I,I)=T(M,1,2)*DX(I,I)+T(M,2,2)*DY(I,I)+T(M,3,2)*DZ(I,I)
        DZO(I,I)=T(M,1,3)*DX(I,I)+T(M,2,3)*DY(I,I)+T(M,3,3)*DZ(I,I)
        DXO(I,L)=T(M,1,1)*DX(I,L)+T(M,2,1)*DY(I,L)+T(M,3,1)*DZ(I,L)
        DYO(I,L)=T(M,1,2)*DX(I,L)+T(M,2,2)*DY(I,L)+T(M,3,2)*DZ(I,L)
        DZO(I,L)=T(M,1,3)*DX(I,L)+T(M,2,3)*DY(I,L)+T(M,3,3)*DZ(I,L)
        DO 17 J=1,NB
        IF(I.EQ.J) GO TO 17
        IF(CPS(K,J).EQ.1) GO TO 177
        IF(C(K,I).EQ.(J-1)) GO TO 17
        DXO(I,J)=DXO(I,L)
        DYO(I,J)=DYO(I,L)
        DZO(I,J)=DZO(I,L)
        GO TO 17
  177   DXO(I,J)=T(M,1,1)*DX(I,J)+T(M,2,1)*DY(I,J)+T(M,3,1)*DZ(I,J)
        DYO(I,J)=T(M,1,2)*DX(I,J)+T(M,2,2)*DY(I,J)+T(M,3,2)*DZ(I,J)
        DZO(I,J)=T(M,1,3)*DX(I,J)+T(M,2,3)*DY(I,J)+T(M,3,3)*DZ(I,J)
  17    CONTINUE
        DO 367 I=1,NB
        DXO(I,I)=DX(I,I)
        DYO(I,I)=DY(I,I)
  367   DZO(I,I)=DZ(I,I)
C
C       COMPUTE TOTAL EXTERNAL FORCE ON VEHICLE (IN REF. COORD.)
C
        FTXO=0.
        FTYO=0.
        FTZO=0.
        DO 247 N=1,NB
        FTXO=FTXO+FEXO(N)
        FTYO=FTYO+FEYO(N)
  247   FTZO=FTZO+FEZO(N)
C
C       ADDITIONAL AUGMENTED INERTIA DYADICS (IN REF.BODY FRAME)
C
        DO 37 I=1,NB
        DO 37 J=1,NB
        IF(I.GE.J) GO TO 37
        DX2=DXO(I,J)*DXO(J,I)
        DY2=DYO(I,J)*DYO(J,I)
        DZ2=DZO(I,J)*DZO(J,I)
        PS(I,J,1,1)=-TM*(DY2+DZ2)
        PS(I,J,1,2)=TM*DXO(J,I)*DYO(I,J)
        PS(I,J,1,3)=TM*DXO(J,I)*DZO(I,J)
        PS(I,J,2,1)=TM*DYO(J,I)*DXO(I,J)
        PS(I,J,2,2)=-TM*(DX2+DZ2)
        PS(I,J,2,3)=TM*DYO(J,I)*DZO(I,J)
        PS(I,J,3,1)=TM*DZO(J,I)*DXO(I,J)
        PS(I,J,3,2)=TM*DZO(J,I)*DYO(I,J)
        PS(I,J,3,3)=-TM*(DX2+DY2)
        DO 378 M=1,3
        DO 378 N=1,3
  378   PS(J,I,M,N)=PS(I,J,N,M)
```

```
 37   CONTINUE
      DO 751 M=1,3
      DO 751 N=1,3
751   PS(1,1,M,N)=PH(1,M,N)
C
C     TRANSFORM AUGMENTED BODY INERTIA DYADICS TO REF. BODY FRAME
C
      DO 363 I=2,NB
      M=HI(I)
      DO 364 J=1,3
      DO 364 K=1,3
      AB(J,K)=0.
      DO 364 L=1,3
      AB(J,K)=AB(J,K)+PH(I,J,L)*T(M,L,K)
364   CONTINUE
      DO 365 J=1,3
      DO 365 K=1,3
      PS(I,I,J,K)=0.
      DO 365 L=1,3
      PS(I,I,J,K)=PS(I,I,J,K)+T(M,L,J)*AB(L,K)
365   CONTINUE
363   CONTINUE
C
C     COMPUTE THE PGSO VECTORS FOR EACH FLEX. APPENDAGE
C
      DO 208 K=1,NF
      KK=F(K,1)+1
      M=HI(KK)
      JNT=F(K,3)
      IF(KK.EQ.1) GO TO 2090
      DO 209 I=1,3
      PGSO(K,I)=0.
      DO 209 J=1,3
209   PGSO(K,I)=PGSO(K,I)+T(M,J,I)*(-MCK(K,J))
      GO TO 208
2090  CONTINUE
      DO 2091 I=1,3
2091  PGSO(K,I)=-MCK(K,I)
208   CONTINUE
C
C     VECTOR CROSS PRODUCTS DESCRIBING SYSTEM ROTATIONAL COUPLING.
C       (QUADRATIC TERMS INVOLVING THE CONNECTING BODY ANGULAR
C       VELOCITIES AND THE MUTUAL BARYCENTER-HINGE VECTORS)
C
      DO 230 N=1,NB
      I=FI(N)
      DO 476 J=1,3
476   CWWD(N,J)=0.
      CPX=0.
      CPY=0.
      CPZ=0.
      CPFX=0.
      CPFY=0.
      CPFZ=0.
      DCPX=0.
      DCPY=0.
      DCPZ=0.
      DO 2301 L=1,NB
      IL=FI(L)
      IF(IL.NE.0) GO TO 7149
      WDX=WYO(L)*DZO(L,N)-WZO(L)*DYO(L,N)
      WDY=WZO(L)*DXO(L,N)-WXO(L)*DZO(L,N)
      WDZ=WXO(L)*DYO(L,N)-WYO(L)*DXO(L,N)
      WWFDX=WYO(L)*WDZ-WZO(L)*WDY
      WWFDY=WZO(L)*WDX-WXO(L)*WDZ
      WWFDZ=WXO(L)*WDY-WYO(L)*WDX
      GO TO 7148
7149  CONTINUE
      WWFDX=0.
```

```
      WWFDY=0.
      WWFDZ=0.
 7148 CONTINUE
      IF(I.EQ.0) GO TO 482
      CWWD(N,1)=CWWD(N,1)+WWFDX
      CWWD(N,2)=CWWD(N,2)+WWFDY
      CWWD(N,3)=CWWD(N,3)+WWFDZ
  482 CONTINUE
      CPFX=CPFX+WWFDX
      CPFY=CPFY+WWFDY
      CPFZ=CPFZ+WWFDZ
      IF(N.EQ.L) GO TO 2301
      WWDX=TM*WWFDX+FEXO(L)
      WWDY=TM*WWFDY+FEYO(L)
      WWDZ=TM*WWFDZ+FEZO(L)
      DWWDX=DYO(N,L)*WWDZ-DZO(N,L)*WWDY
      DWWDY=DZO(N,L)*WWDX-DXO(N,L)*WWDZ
      DWWDZ=DXO(N,L)*WWDY-DYO(N,L)*WWDX
      CPX=CPX+DWWDX
      CPY=CPY+DWWDY
      CPZ=CPZ+DWWDZ
 2301 CONTINUE
      DFX=DYO(N,N)*FEZO(N)-DZO(N,N)*FEYO(N)
      DFY=DZO(N,N)*FEXO(N)-DXO(N,N)*FEZO(N)
      DFZ=DXO(N,N)*FEYO(N)-DYO(N,N)*FEXO(N)
      IF(I.NE.0) GO TO 7147
      HX=PS(N,N,1,1)*WXO(N)+PS(N,N,1,2)*WYO(N)+PS(N,N,1,3)*WZO(N)
      HY=PS(N,N,2,1)*WXO(N)+PS(N,N,2,2)*WYO(N)+PS(N,N,2,3)*WZO(N)
      HZ=PS(N,N,3,1)*WXO(N)+PS(N,N,3,2)*WYO(N)+PS(N,N,3,3)*WZO(N)
      GO TO 7146
 7147 CONTINUE
      HX=0.
      HY=0.
      HZ=0.
 7146 CONTINUE
      IF(I.EQ.0) GO TO 243
      FACT=MSB(N)/TM
      FTXM=FTXO*FACT
      FTYM=FTYO*FACT
      FTZM=FTZO*FACT
      PGFX=(PGSO(I,2)*(FEZO(N)-FTZM)-PGSO(I,3)*(FEYO(N)-FTYM))/MSB(N)
      PGFY=(PGSO(I,3)*(FEXO(N)-FTXM)-PGSO(I,1)*(FEZO(N)-FTZM))/MSB(N)
      PGFZ=(PGSO(I,1)*(FEYO(N)-FTYM)-PGSO(I,2)*(FEXU(N)-FTXM))/MSB(N)
      PWWDX=PGSO(I,2)*CPFZ-PGSO(I,3)*CPFY
      PWWDY=PGSO(I,3)*CPFX-PGSO(I,1)*CPFZ
      PWWDZ=PGSO(I,1)*CPFY-PGSO(I,2)*CPFX
      GO TO 244
  243 CONTINUE
      PGFX=0.
      PGFY=0.
      PGFZ=0.
      PWWDX=0.
      PWWDY=0.
      PWWDZ=0.
  244 CONTINUE
      K = 3*(N-1)
      E(K+1,1)=HY*WZO(N)-HZ*WYO(N)+TXO(N)+CPX+DFX+PGFX-PWWDX
      E(K+2,1)=HZ*WXO(N)-HX*WZO(N)+TYO(N)+CPY+DFY+PGFY-PWWUY
      E(K+3,1)=HX*WYO(N)-HY*WXO(N)+TZO(N)+CPZ+DFZ+PGFZ-PWWDZ
  230 CONTINUE
C
C     A00 MATRIX ELEMENT COMPUTATION (3X3)
C
      DO 3001 I=1,3
      DO 3001 J=1,3
 3001 A00(I,J)=0.
      DO 3 I=1,NB
      DO 3 J=1,NB
      A00(1,1)=A00(1,1)+PS(I,J,1,1)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      A00(1,2)=A00(1,2)+PS(I,J,1,2)
      A00(1,3)=A00(1,3)+PS(I,J,1,3)
      A00(2,2)=A00(2,2)+PS(I,J,2,2)
      A00(2,3)=A00(2,3)+PS(I,J,2,3)
      A00(3,3)=A00(3,3)+PS(I,J,3,3)
    3 CONTINUE
      A00(2,1)=A00(1,2)
      A00(3,1)=A00(1,3)
      A00(3,2)=A00(2,3)
C
C     FLEX. APPEND. CONTRIBUTION TO A00 MATRIX COMPUTATION (3X3)
C
      DO 210 K=1,NB
      KK=FI(K)
      DO 210 L=1,NB
      IF(K.GT.L) GO TO 210
      DO 2103 I=1,3
      DO 2103 J=1,3
 2103 PSF(K,L,I,J)=0.
      LL=FI(L)
      IF(KK.EQ.0) GO TO 2101
      DP1=PGSO(KK,1)*DXO(L,K)
      DP2=PGSO(KK,2)*DYO(L,K)
      DP3=PGSO(KK,3)*DZO(L,K)
      PSF(K,L,1,1)=-DP2-DP3
      PSF(K,L,2,2)=-DP1-DP3
      PSF(K,L,3,3)=-DP1-DP2
      PSF(K,L,1,2)=PGSO(KK,2)*DXO(L,K)
      PSF(K,L,1,3)=PGSO(KK,3)*DXO(L,K)
      PSF(K,L,2,1)=PGSO(KK,1)*DYO(L,K)
      PSF(K,L,2,3)=PGSO(KK,3)*DYO(L,K)
      PSF(K,L,3,1)=PGSO(KK,1)*DZO(L,K)
      PSF(K,L,3,2)=PGSO(KK,2)*DZO(L,K)
 2101 CONTINUE
      IF(LL.EQ.0) GO TO 210
      IF(K.EQ.L) GO TO 2102
      PD1=PGSO(LL,1)*DXO(K,L)
      PD2=PGSO(LL,2)*DYO(K,L)
      PD3=PGSO(LL,3)*DZO(K,L)
      PSF(K,L,1,1)=PSF(K,L,1,1)-PD2-PD3
      PSF(K,L,2,2)=PSF(K,L,2,2)-PD1-PD3
      PSF(K,L,3,3)=PSF(K,L,3,3)-PD1-PD2
      PSF(K,L,1,2)=PSF(K,L,1,2)+DYO(K,L)*PGSO(LL,1)
      PSF(K,L,1,3)=PSF(K,L,1,3)+DZO(K,L)*PGSO(LL,1)
      PSF(K,L,2,1)=PSF(K,L,2,1)+DXO(K,L)*PGSO(LL,2)
      PSF(K,L,2,3)=PSF(K,L,2,3)+DZO(K,L)*PGSO(LL,2)
      PSF(K,L,3,1)=PSF(K,L,3,1)+DXO(K,L)*PGSO(LL,3)
      PSF(K,L,3,2)=PSF(K,L,3,2)+DYO(K,L)*PGSO(LL,3)
      GO TO 210
 2102 CONTINUE
      DO 214 I=1,3
      DO 214 J=1,3
  214 AB(I,J)=PSF(K,L,I,J)
      DO 215 I=1,3
      DO 215 J=1,3
  215 PSF(K,L,I,J)=AB(I,J)+AB(J,I)
  210 CONTINUE
      DO 2151 K=1,NB
      DO 2151 L=1,NB
      IF(K.LE.L) GO TO 2151
      DO 2141 I=1,3
      DO 2141 J=1,3
 2141 PSF(K,L,I,J)=PSF(L,K,J,I)
 2151 CONTINUE
      DO 3004 K=1,NB
      KK=FI(K)
      DO 3004 L=1,NB
      LL=FI(L)
      IF((KK.EQ.0).AND.(LL.EQ.0)) GO TO 3004
```

```
      DO 3003 I=1,3
      DO 3003 J=1,3
      AOO(I,J)=AOO(I,J)-PSF(K,L,I,J)
3003  CONTINUE
3004  CONTINUE
C
C     AOK VECTOR ELEMENT COMPUTATION (3X1)
C
C     AKM SCALAR ELEMENT COMPUTATION
C
      DO 14 M=1,NH
      IQ=H(M)+1
      AV(M,1)=0.
      AV(M,2)=0.
      AV(M,3)=0.
      DO 7 J=1,NB
      DO 7 I=IQ,NB
      DO 11 N=1,3
      IF(EPS(M,I).EQ.0) GO TO 7
      PSG(J,I,N)=0.
      DO 10 L=1,3
10    PSG(J,I,N)=PSG(J,I,N)+(PS(J,I,N,L)-PSF( J,I,N,L))*GO(M,L)
11    AV(M,N)=AV(M,N)+PSG(J,I,N)
7     CONTINUE
      DO 14 K=1,NH
      IF(K.GT.M) GO TO 14
      JQ=H(K)+1
      AIS(1)=0.
      AIS(2)=0.
      AIS(3)=0.
      DO 15 J=JQ,NB
      DO 15 I=IQ,NB
      IF((EPS(K,J).EQ.0).OR.(EPS(M,I).EQ.0)) GO TO 15
      DO 18 N=1,3
18    AIS(N)=AIS(N)+PSG(J,I,N)
15    CONTINUE
      AS(K,M)=GO(K,1)*AIS(1)+GO(K,2)*AIS(2)+GO(K,3)*AIS(3)
14    CONTINUE
C
C     AOF MATRIX (3 X NKT) (REF. BODY/FLEX. APPENDAGE COUPLING)
C
      DO 219 K=1,NF
      JK=F(K,3)
      JQ=F(K,1)+1
      DO 222 I=1,3
      DO 222 J=1,3
222   AB(I,J)=0.
      DO 221 L=1,NB
      AB(1,2)=AB(1,2)-DZO(L,JQ)
      AB(1,3)=AB(1,3)+DYO(L,JQ)
221   AB(2,3)=AB(2,3)-DXO(L,JQ)
      AB(2,1)=-AB(1,2)
      AB(3,1)=-AB(1,3)
      AB(3,2)=-AB(2,3)
      DO 220 I=1,3
      DO 220 J=1,JK
      AOF(K,I,J)=DLKO(K,I,J)
      DO 220 L=1,3
220   AOF(K,I,J)=AOF(K,I,J) - AB(I,L)*PKO(K,L,J)
219   CONTINUE
C
C     AKF VECTOR (1 X NKT) (FLEX. COUPLING WITH RIGID SUBSTRUCTURES)
C
      DO 224 K=1,NF
      JK=F(K,3)
      JQ=F(K,1)+1
      DO 2245 J=1,JK
2245  ZSR(K,J)=0.
      DO 224 M=1,NH
      DO 231 I=1,3
```

```
      DO 231 J=1,3
  231 AB(I,J)=0.
      DO 226 L=1,NB
      IF(EPS(M,L).EQ.0) GO TO 226
      AB(1,2)=AB(1,2)-DZO(L,JQ)
      AB(1,3)=AB(1,3)+DYO(L,JQ)
      AB(2,3)=AB(2,3)-DXO(L,JQ)
  226 CONTINUE
      AB(2,1)=-AB(1,2)
      AB(3,1)=-AB(1,3)
      AB(3,2)=-AB(2,3)
      DO 228 I=1,3
      DO 228 J=1,JK
      DUR(I,J)=DLKO(K,I,J)
      IF(EPS(M,K).EQ.0) DUR(I,J)=0.
      DO 228 L=1,3
  228 DUR(I,J)=DUR(I,J)-AB(I,L)*PKO(K,L,J)
      DO 2241 J=1,JK
      DO 2241 I=1,3
 2241 ZSR(K,J)=ZSR(K,J)+DUR(I,J)*WGJ(M,I)
      DO 229 J=1,JK
      AKF(K,M,J)=0.
      DO 229 I=1,3
  229 AKF(K,M,J)=AKF(K,M,J)+GO(M,I)*DUR(I,J)
  224 CONTINUE
C
C     COMPUTE CORRECTION ELEMENTS FOR (E) VECTOR
C
      DO 41 J=2,NB
      JK=MI(J)
      DO 411 M=1,3
  411 CW(J,M)=0.
      DO 42 K=1,JK
      IF(EPS(K,J).EQ.0) GO TO 42
      CW(J,1)=CW(J,1)+WGJ(K,1)
      CW(J,2)=CW(J,2)+WGJ(K,2)
      CW(J,3)=CW(J,3)+WGJ(K,3)
   42 CONTINUE
   41 CONTINUE
      DO 40 I=1,NB
      EA(1)=0.
      EA(2)=0.
      EA(3)=0.
      DO 401 J=2,NB
      DO 4507 M=1,3
      DO 4507 L=1,3
 4507 EA(M)=EA(M)+(PS(I,J,M,L)-PSF(I,J,M,L))*CW(J,L)
  401 CONTINUE
      KI=3*(I-1)
      E(KI+1,1)=E(KI+1,1)-EA(1)
      E(KI+2,1)=E(KI+2,1)-EA(2)
      E(KI+3,1)=E(KI+3,1)-EA(3)
   40 CONTINUE
      DO 55 MI=1,3
   55 EC(MI)=E(MI,1)
      DO 52 J=2,NB
      DO 52 M=1,3
      KI=3*(J-1)+M
   52 EC(M)=EC(M)+E(KI,1)
      I=0
      DO 60 K=1,NM
      JK=M(K)+1
      IF(PI(K).NE.0) GO TO 60
      I=I+1
      EC(I+3)=0.
      DO 601 M=1,3
  601 CE(M)=0.
      DO 61 J=JK,NB
      IF(EPS(K,J).EQ.0) GO TO 61
```

```
      DO 65 M=1,3
      J1=3*(J-1)+M
   65 CE(M)=CE(M)+E(J1,I)
   61 CONTINUE
      DO 66 L=1,3
   66 EC(I+3)=EC(I+3)+GO(K,L)*CE(L)
      EC(I+3)=EC(I+3)+TH(K)
   60 CONTINUE
      DO 610 I=1,3
      DO 610 J=1,NH
      IF(PI(J).EQ.0) GO TO 610
      EC(I)=EC(I)-AV(J,I)*GMDD(J)
  610 CONTINUE
      K=0
      IV=3
      DO 612 I=1,NH
      IF(PI(I).NE.0) GO TO 612
      K=K+1
      IV=IV+1
      DO 611 J=1,NH
      IF(PI(J).EQ.0) GO TO 611
      IF(I.GT.J) AS(I,J)=AS(J,I)
      EC(K+3)=EC(K+3)-AS(I,J)*GMDD(J)
  611 CONTINUE
  612 CONTINUE
C
C     COMPUTE RT. HAND SIDE OF APPENDAGE EQUATIONS (IN APPEND. COORDS.)
C
      DO 483 K=1,NF
      I=F(K,1)+1
      M=HI(I)
      CQ(1)=FTXO/TM + CWWD(I,1)
      CQ(2)=FTYO/TM + CWWD(I,2)
      CQ(3)=FTZO/TM + CWWD(I,3)
      IF(I.EQ.1) GO TO 4840
      DO 484 J=1,3
      VE(K,J)=0.
      DO 484 L=1,3
  484 VE(K,J)=VE(K,J)+T(M,J,L)*CQ(L)
      GO TO 483
 4840 CONTINUE
      DO 4841 J=1,3
 4841 VE(K,J)=CQ(J)
  483 CONTINUE
      DO 485 K=1,NF
      NL=F(K,2)
      DO 486 N=1,NL
      N6=6*(N-1)
      DO 488 J=1,3
      JN=N6+J
      JM=JN+3
      VB(K,JN)=FF(K,N,J)
  488 VB(K,JM)=TF(K,N,J)
  486 CONTINUE
  485 CONTINUE
      NV=IV
      DO 491 K=1,NF
      JN=F(K,3)
      NL=F(K,2)
      NL6=6*NL
      DO 492 J=1,JN
      IL=NV+J
      VV1=-WF(K,J)*(2.*ZF(K,J)*ETD(K,J)+WF(K,J)*ET(K,J))
      DO 493 N=1,NL6
  493 VV1=VV1+EIG(K,N,J)*VB(K,N)
      DO 494 N=1,3
  494 VV1=VV1-PK(K,N,J)*VE(K,N)
      VV1=VV1-ZSR(K,J)
      EC(IL)=VV1
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      DO 4920 L=1,NH
      IF(PI(L).EQ.0) GO TO 4920
      EC(IL)=EC(IL)-AKF(L,K,J)*GHDD(L)
4920  CONTINUE
492   CONTINUE
491   NV=NV+JN
C
C     ENTER CONSTANTS INTO FLEX. BODY PORTION OF COEFF. MATRIX A
C
      NV=IV
      DO 462 K=1,NF
      NL=F(K,3)
      DO 463 I=1,NL
      IL=NV+I
      DO 463 J=1,NL
      JL=NV+J
      A(IL,JL)=0.
      IF(I.EQ.J) A(IL,JL)=1.
463   CONTINUE
462   NV=NV+NL
C
C     ENTER COEFF. WHICH COUPLE REF. BODY AND FLEX. APPENDAGES INTO A
C
      NV=IV
      DO 464 K=1,NF
      NL=F(K,3)
      DO 465 J=1,3
      DO 465 I=1,NL
      IL=NV+I
      A(IL,J)=AJF(K,J,I)
465   A(J,IL)=A(IL,J)
464   NV=NV+NL
C
C     ENTER COEFF. WHICH COUPLE SUBSTR. BODIES AND FLEX. APPEND. INTO A
C
      NV=IV
      DO 466 K=1,NF
      NL=F(K,3)
      JI=0
      DO 467 J=1,NH
      IF(PI(J).NE.0) GO TO 467
      JI=JI+1
      DO 4671 I=1,NL
      IL=NV+I
      A(IL,JI+3)=AKF(K,J,I)
      A(JI+3,IL)=A(IL,JI+3)
4671  CONTINUE
467   CONTINUE
466   NV=NV+NL
C
C     CALCULATE FLEX. BODY COUPLING COEFF. AND ENTER INTO A MATRIX
C
      NCO=IV
      DO 473 L=1,NF
      NL=F(L,3)
      NRO=IV
      DO 474 K=1,NF
      NR=F(K,3)
      IF(K.EQ.L) GO TO 474
      DO 475 I=1,NR
      IK=NRO+I
      DO 475 J=1,NL
      JK=NCO+J
      A(IK,JK)=0.
      DO 4750 N=1,3
      A(IK,JK)=A(IK,JK)-PKO(K,N,I)*PKO(L,N,J)/TM
4750  CONTINUE
      A(JK,IK)=A(IK,JK)
475   CONTINUE
474   NRO=NRO+NR
```

```
473   NCO=NCO+NL
C
C     LOAD SYSTEM MATRIX (A) WITH A00,A0K,AKM ELEMENTS
C
      DO 23 I=1,3
      DO 23 J=1,3
23    A(I,J)=A00(I,J)
      DO 24 I=1,3
      K=0
      DO 24 J=1,NH
      IF(PI(J).NE.0) GO TO 24
      K=K+1
      A(K+3,I)=AV(J,I)
      A(I,K+3)=AV(J,I)
24    CONTINUE
      K=0
      DO 250 I=1,NH
      IF(PI(I).NE.0) GO TO 250
      K=K+1
      L=0
      DO 25 J=1,NH
      IF(PI(J).NE.0) GO TO 25
      L=L+1
      IF(K.GT.L) GO TO 26
      A(K+3,L+3)=AS(I,J)
      GO TO 25
26    A(K+3,L+3)=A(L+3,K+3)
25    CONTINUE
250   CONTINUE
C
C     ANGULAR MOMENTUM OF THE SYSTEM
C
      IF(PI(NH+1).NE.1) GO TO 8752
      DO 5651 I=1,3
      HH(I)=0.
      DO 5651 J=1,3
5651  HH(I)=HH(I)+A(I,J)*W0(J)
      DO 5652 I=1,3
      DO 5652 J=1,QH
5652  HH(I)=HH(I)+AV(J,I)*GMD(J)
      DO 5653 I=1,3
      DO 5653 K=1,NF
      NL=F(K,3)
      DO 5654 J=1,NL
5654  HH(I)=HH(I)+A0F(K,I,J)*ETD(K,J)
5653  CONTINUE
      HM=SQRT(HH(1)**2 + HH(2)**2 + HH(3)**2)
8752  CONTINUE
C
C     SOLVE SYSTEM MATRIX FOR REFERENCE BODY ANG. ACCELERATION AND HINGE
C                  (RELATIVE) ROTATIONAL ACCELERATIONS
C
      NT=V+NTMO
      IT=IV+NTMO
      KV=IV
      CALL CHOLD(592,A,ST,IT,EC,0.,1.D-7)
      DO 910 J=NT,4,-1
      IF(J.LE.V) GO TO 913
      JV=J-(V-IV)
      EC(J)=EC(JV)
      GO TO 910
913   CONTINUE
      K=J-3
      IF(PI(K).NE.0) GO TO 911
      EC(J)=EC(KV)
      KV=KV-1
      GO TO 910
911   EC(J)=GMDD(K)
910   CONTINUE
```

```
      DO 9003 I=1,V
9003  WDOT(I)=EC(I)
      I=V
      DO 9001 K=1,NF
      NL=F(K,3)
      DO 9002 N=1,NL
      IO=I+N
9002  ETDD(K,N)=EC(IO)
9001  I=I+NL
92    CONTINUE
      RETURN
      END
```

DIAGNOSTICS

ATION TIME = 30.73 SUPS

CSSL.TRAN.CSSL

Appendix E

Subroutine MBDYFL Listing and User Requirements

Subroutine Entry Statements

Same as MBDYFN (see Appendix D)

Input / Output Variable Type and Storage Specifications

Same as MBDYFN (see Appendix D)

External Subroutines Called

AINVD—double precision matrix inversion subroutine. Inverts any real square, nonsingular matrix, A, and leaves the result in A (see statement 419).

Subroutine Setup

Same as MBDYFN (see Appendix D)

Data Restrictions

Same as MBDYFN

Core Storage Required

Code: 3500 words

Data: ~500 words (minimum; varies with n, n_f)

Listing

```
 1.       SUBROUTINE MBDYFL(NC,C,MB,MA,PB,PA,G,PI,NF,F,EIG,REC,RF,WF,ZF)
 2.  C
 3.  C    ADJUSTABLE DIMENSIONS
 4.  C
 5.       INTEGER PI(1),C(NC,2)
 6.       REAL MB(1),MA(NC,7),PB(NC,3),PA(NC,NC,3)
 7.       PARAMETER QC=1,QH=2,QF=1,NK=1,NKT=7
 8.       PARAMETER N6K=6*NK,S=QC+1,V=QH+3,V4=4*V,S3=3*S,Q=QH,NH=QH
 9.       PARAMETER ST=V+QF*NKT,S4=4*ST
10.  C
11.  C    ADDITIONAL DIMENSIONED VARIABLES
12.  C
13.       DOUBLE PRECISION A(ST,ST),WRK(S4),BMASS(S)
14.       INTEGER EPS(Q,S),CPS(QC,S),H(Q),HI(S),FI(S),F(NF,3)
15.       REAL A00(3,3),AB(3,3),A0F(QF,3,NKT),AKF(QF,QH,NKT),AS(Q,Q),AV(Q,3)
16.      $,AIS(3),CE(3),CK(QF,3),CW(3),DX(S,S),DY(S,S),DZ(S,S),DXO(S,S),DYO(
17.      $S,S),DZO(S,S),DLK(QF,3,NKT),DUR(3,NKT),EIG(NF,N6K,NKT),FEXO(S),FEY
18.      $O(S),FEZO(S),FS(S,3),GO(Q,3),G(Q,3),IXX(S),IYY(S),IZZ(S),IXY(S),IX
19.      $Z(S),IYZ(S),LX(S,S),LY(S,S),LZ(S,S),MSB(S),MCK(QF,3),PH(S,3,3),PSG
20.      $(S,S,3),PS(S,S,3,3),PK(QF,3,NKT),PGSO(QF,3),RF(NF,NK,3),REC(NF,6,N
21.      $KT),TXO(S),TYO(S),TZO(S),T(Q,3,3),TS(S,3),U(QF,NK,3),VE(QF,3),VB(Q
22.      $F,N6K),WF(NF,NKT),ZF(NF,NKT),HM(3)
```

```
      EQUIVALENCE (A,PS),(LX,DXO),(LY,DYO),(LZ,DZO)
      NB=NC+1
C
C     DEFINE EPS(K,J) USING C
C
      DO 86 K=1,NC
      DO 86 J=2,NB
      IF(K.EQ.(J-1)) CPS(K,J)=1
      IF(K.LT.(J-1)) GO TO 87
      GO TO 86
 87   CONTINUE
      J0=K+1
      J1=J-1
      DO 89 L=J0,J1
      IF(K.GT.(L-1)) GO TO 89
      IF((CPS(K,L).EQ.1).AND.(C(J-1,1).EQ.(L-1))) CPS(K,J)=1

 89   CONTINUE
 86   CONTINUE
      L=0
      DO 1 J=1,NC
      KK=C(J,2)
      DO 1 K=1,KK
      L=L+1
      DO 1 I=1,NB
 1    EPS(L,I)=CPS(J,I)
C
C     COMPUTE H(I)=C, WHERE I=HINGE LABEL AND C=CONNECTION LABEL
C
      I=0
      DO 8 J=2,NB
      KK=C(J-1,2)
      DO 8 K=1,KK
      I=I+1
 8    H(I)=J-1
C
C     COMPUTE HI(I)=J, WHERE I=BODY LABEL+1 AND J=NEAREST HINGE LABEL
C
      HI(1)=1
      HI(NB)=NH
      DO 47 I=NH,1
      IF(I.EQ.1) GO TO 47
      K1=H(I)
      K2=H(I-1)
      IF(K1.EQ.K2) GO TO 47
      HI(K2+1)=I-1
 47   CONTINUE
C
C     DEFINE FI(J)=K, WHERE J=BODY-LABEL+1 AND K IS APPENDAGE-LABEL
C                   (IF K=0, BODY HAS NO FLEX. APPENDAGE)
C
      DO 239 N=1,NB
 239  FI(N)=0
      DO 242 K=1,NF
      JN=F(K,1)+1
 242  FI(JN)=K
      NF=NF
      NB=NB
C
C     DEFINE SUBSTRUCTURE MASSES
C
      MSB(1)=MB(7)
      DO 248 N=2,NB
 248  MSB(N)=MA(N-1,7)
C
C     TOTAL NUMBER OF FLEX. APPENDAGE MODES TO BE RETAINED
C
      NTMO=0
      DO 461 K=1,NF
```

```
461   NTHO=NTHO+F(K,3)
C
C     INITIAL CALCULATION OF BARYCENTER VECTORS W.R.T. BODY C.G.S
C                          AND HINGE POINTS
C
      IXX(1)=MB(1)
      IYY(1)=MB(2)
      IZZ(1)=MB(3)
      IXY(1)=MB(4)
      IXZ(1)=MB(5)
      IYZ(1)=MB(6)
      BMASS(1)=MB(7)
      TM=BMASS(1)
      DO 35 J=2,NB
      IXX(J)=MA(J-1,1)
      IYY(J)=MA(J-1,2)
      IZZ(J)=MA(J-1,3)
      IXY(J)=MA(J-1,4)
      IXZ(J)=MA(J-1,5)
      IYZ(J)=MA(J-1,6)
      BMASS(J)=MA(J-1,7)
35    TM=TM+BMASS(J)
      DO 149 I=1,NB
      I1=I-1
      DO 149 J=1,NB
      J1=J-1
      IF(I.EQ.J) GO TO 163
      IF(I.GT.J) GO TO 70
      IF(I.EQ.1) GO TO 80
      IF(CPS(I1,J).EQ.1) GO TO 400
70    LX(I,J)=PA(I1,I1,1)
      LY(I,J)=PA(I1,I1,2)
      LZ(I,J)=PA(I1,I1,3)
      GO TO 149
400   CONTINUE
      DO 600 K=1,J1
      IF(CPS(K,J).EQ.1) GO TO 500
600   CONTINUE
      GO TO 149
500   LX(I,J)=PA(I1,K,1)
      LY(I,J)=PA(I1,K,2)
      LZ(I,J)=PA(I1,K,3)
      GO TO 149
80    DO 90 L=1,J1
      IF(CPS(L,J).EQ.1) GO TO 101
90    CONTINUE
      GO TO 149
101   LX(I,J)=PB(L,1)
      LY(I,J)=PB(L,2)
      LZ(I,J)=PB(L,3)
      GO TO 149
163   LX(I,J)=0.
      LY(I,J)=0.
      LZ(I,J)=0.
149   CONTINUE
      DO 13 N=1,NB
      DO 13 J=1,NB
      DX(N,J)=LX(N,J)
      DY(N,J)=LY(N,J)
      DZ(N,J)=LZ(N,J)
      DO 13 K=1,NB
      DX(N,J)=DX(N,J)-(BMASS(K)/TM)*LX(N,K)
      DY(N,J)=DY(N,J)-(BMASS(K)/TM)*LY(N,K)
13    DZ(N,J)=DZ(N,J)-(BMASS(K)/TM)*LZ(N,K)
C
C     CALCULATION OF AUGMENTED INERTIA DYADICS FOR EACH BODY
C
      DO 31 N=1,NB
      PH(N,1,1)=IXX(N)
```

```
        PH(N,1,2)=-IXY(N)
        PH(N,1,3)=-IXZ(N)
        PH(N,2,2)=IYY(N)
        PH(N,2,3)=-IYZ(N)
        PH(N,3,3)=IZZ(N)
        DO 30 J=1,NB
        PH(N,1,1)=PH(N,1,1)+BMASS(J)*(DY(N,J)**2+DZ(N,J)**2)
        PH(N,1,2)=PH(N,1,2)-BMASS(J)*DX(N,J)*DY(N,J)
        PH(N,1,3)=PH(N,1,3)-BMASS(J)*DX(N,J)*DZ(N,J)
        PH(N,2,2)=PH(N,2,2)+BMASS(J)*(DX(N,J)**2+DZ(N,J)**2)
        PH(N,2,3)=PH(N,2,3)-BMASS(J)*DY(N,J)*DZ(N,J)
30      PH(N,3,3)=PH(N,3,3)+BMASS(J)*(DX(N,J)**2+DY(N,J)**2)
        PH(N,2,1)=PH(N,1,2)
        PH(N,3,1)=PH(N,1,3)
31      PH(N,3,2)=PH(N,2,3)
C
C       ADDITIONAL AUGMENTED INERTIA DYADICS (IN REF.BODY FRAME)
C
        DO 751 J=1,NB
        DO 751 M=1,3
        DO 751 N=1,3
751     PS(J,J,M,N)=PH(J,M,N)
        DO 37 I=1,NB
        DO 37 J=1,NB
        IF(I.GE.J) GO TO 37
        DX2=DX(I,J)*DX(J,I)
        DY2=DY(I,J)*DY(J,I)
        DZ2=DZ(I,J)*DZ(J,I)
        PS(I,J,1,1)=-TM*(DY2+DZ2)
        PS(I,J,1,2)=TM*DX(J,I)*DY(I,J)
        PS(I,J,1,3)=TM*DX(J,I)*DZ(I,J)
        PS(I,J,2,1)=TM*DY(J,I)*DX(I,J)
        PS(I,J,2,2)=-TM*(DX2+DZ2)
        PS(I,J,2,3)=TM*DY(J,I)*DZ(I,J)
        PS(I,J,3,1)=TM*DZ(J,I)*DX(I,J)
        PS(I,J,3,2)=TM*DZ(J,I)*DY(I,J)
        PS(I,J,3,3)=-TM*(DX2+DY2)
        DO 378 M=1,3
        DO 378 N=1,3
378     PS(J,I,M,N)=PS(I,J,N,M)
37      CONTINUE
C
C       A00 MATRIX ELEMENT COMPUTATION (3X3)
C
        DO 3001 I=1,3
        DO 3001 J=1,3
3001    A00(I,J)=0.
        DO 3 I=1,NB
        DO 3 J=1,NB
        A00(1,1)=A00(1,1)+PS(I,J,1,1)
        A00(1,2)=A00(1,2)+PS(I,J,1,2)
        A00(1,3)=A00(1,3)+PS(I,J,1,3)
        A00(2,2)=A00(2,2)+PS(I,J,2,2)
        A00(2,3)=A00(2,3)+PS(I,J,2,3)
        A00(3,3)=A00(3,3)+PS(I,J,3,3)
3       CONTINUE
        A00(2,1)=A00(1,2)
        A00(3,1)=A00(1,3)
        A00(3,2)=A00(2,3)
C
C       A0K VECTOR ELEMENT COMPUTATION (3X1)
C
C       AKM SCALAR ELEMENT COMPUTATION
C
        DO 14 M=1,NH
        IQ=H(M)+1
        AV(M,1)=0.
        AV(M,2)=0.
        AV(M,3)=0.
```

```
      DO 7 J=1,NB
      DO 7 I=IQ,NB
      DO 11 N=1,3
      IF(EPS(M,I).EQ.0) GO TO 7
      PSG(J,I,N)=0.
      DO 10 L=1,3
10    PSG(J,I,N)=PSG(J,I,N)+PS(J,I,N,L)*G(M,L)
11    AV(M,N)=AV(M,N)+PSG(J,I,N)
7     CONTINUE
      DO 14 K=1,NM
      IF(K.GT.M) GO TO 14
      JQ=M(K)+1
      AIS(1)=0.
      AIS(2)=0.
      AIS(3)=0.
      DO 15 J=JQ,NB
      DO 15 I=IQ,NB
      IF((EPS(K,J).EQ.0).OR.(EPS(M,I).EQ.0)) GO TO 15
      DO 18 N=1,3
18    AIS(N)=AIS(N)+PSG(J,I,N)
15    CONTINUE
      AS(K,M)=G(K,1)*AIS(1)+G(K,2)*AIS(2)+G(K,3)*AIS(3)
14    CONTINUE
C
C     DEFINE PK(3 X NKT ARRAY)
C     DEFINE DLK-TRANSPOSE MATRIX (3 X NKT ARRAY)
C
      DO 201 K=1,NF
      JNT=F(K,3)
      DO 201 I=1,3
      DO 201 J=1,JNT
      PK(K,I,J)=REC(K,I,J)
201   DLK(K,I,J)=REC(K,I+3,J)
C
C     AOF MATRIX (3 X NKT) (REF. BODY/FLEX. APPENDAGE COUPLING)
C
      DO 219 K=1,NF
      JK=F(K,3)
      JQ=F(K,1)+1
      DO 222 I=1,3
      DO 222 J=1,3
222   AB(I,J)=0.
      DO 221 L=1,NB
      AB(1,2)=AB(1,2)-DZ(L,JQ)
      AB(1,3)=AB(1,3)+DY(L,JQ)
221   AB(2,3)=AB(2,3)-DX(L,JQ)
      AB(2,1)=-AB(1,2)
      AB(3,1)=-AB(1,3)
      AB(3,2)=-AB(2,3)
      DO 220 I=1,3
      DO 220 J=1,JK
      AOF(K,I,J)=DLK(K,I,J)
      DO 220 L=1,3
220   AOF(K,I,J)=AOF(K,I,J) - AB(I,L)*PK(K,L,J)
219   CONTINUE
C
C     AKF VECTOR (1 X NKT) (FLEX. COUPLING WITH RIGID SUBSTRUCTURES)
C
      DO 224 K=1,NF
      JK=F(K,3)
      JQ=F(K,1)+1
      DO 224 M=1,NM
      DO 231 I=1,3
      DO 231 J=1,3
231   AB(I,J)=0.
      DO 226 L=1,NB
      IF(EPS(M,L).EQ.0) GO TO 226
      AB(1,2)=AB(1,2)-DZ(L,JQ)
      AB(1,3)=AB(1,3)+DY(L,JQ)
```

```
      AB(2,3)=AB(2,3)-DX(L,JW)
226   CONTINUE
      AB(2,1)=-AB(1,2)
      AB(3,1)=-AB(1,3)
      AB(3,2)=-AB(2,3)
      DO 228 I=1,3
      DO 228 J=1,JK
      DUR(I,J)=ULK(K,I,J)
      IF(EPS(M,K).EQ.0) DUR(I,J)=0.
      DO 228 L=1,3
228   DUR(I,J)=DUR(I,J)-AB(I,L)*PK(K,L,J)
      DO 229 J=1,JK
      AKF(K,M,J)=0.
      DO 229 I=1,3
229   AKF(K,M,J)=AKF(K,M,J)+G(M,I)*DUR(I,J)
224   CONTINUE
C
C     ENTER CONSTANTS INTO FLEX. BODY PORTION OF COEFF. MATRIX A
C
      IV=3
      DO 6129 I=1,NH
      IF(PI(I).NE.0) GO TO 6129
      IV=IV+1
6129  CONTINUE
      NV=IV
      DO 462 K=1,NF
      NL=F(K,3)
      DO 463 I=1,NL
      IL=NV+I
      DO 463 J=1,NL
      JL=NV+J
      A(IL,JL)=0.
      IF(I.EQ.J) A(IL,JL)=1.
463   CONTINUE
462   NV=NV+NL
C
C     ENTER COEFF. WHICH COUPLE REF. BODY AND FLEX. APPENDAGES INTO A
C
      NV=IV
      DO 464 K=1,NF
      NL=F(K,3)
      DO 465 J=1,3
      DO 465 I=1,NL
      IL=NV+I
      A(IL,J)=AJF(K,J,I)
465   A(J,IL)=A(IL,J)
464   NV=NV+NL
C
C     ENTER COEFF. WHICH COUPLE SUBSTR. BODIES AND FLEX. APPEND. INTO A
C
      NV=IV
      DO 466 K=1,NF
      NL=F(K,3)
      JI=0
      DO 467 J=1,NH
      IF(PI(J).NE.0) GO TO 467
      JI=JI+1
      DO 4671 I=1,NL
      IL=NV+I
      A(IL,JI+3)=AKF(K,J,I)
      A(JI+3,IL)=A(IL,JI+3)
4671  CONTINUE
467   CONTINUE
466   NV=NV+NL
C
C     CALCULATE FLEX. BODY COUPLING COEFF. AND ENTER INTO A MATRIX
C
      NCO=IV
      DO 473 L=1,NF
```

```
      NL=F(L,3)
      NRO=IV
      DO 474 K=1,NF
      NR=F(K,3)
      IF(K.EQ.L) GO TO 474
      DO 475 I=1,NR
      IK=NRO+I
      DO 475 J=1,NL
      JK=NCO+J
      A(IK,JK)=0.
      DO 4750 N=1,3
      A(IK,JK)=A(IK,JK)-PK(K,N,I)*PK(L,N,J)/TM
 4750 CONTINUE
      A(JK,IK)=A(IK,JK)
  475 CONTINUE
  474 NRO=NRO+NR
  473 NCO=NCO+NL
C
C     LOAD SYSTEM MATRIX (A) WITH AOO,AOK,AKM ELEMENTS
C
      DO 23 I=1,3
      DO 23 J=1,3
   23 A(I,J)=AOO(I,J)
      DO 24 I=1,3
      K=0
      DO 24 J=1,NM
      IF(PI(J).NE.0) GO TO 24
      K=K+1
      A(K+3,I)=AV(J,I)
      A(I,K+3)=AV(J,I)
   24 CONTINUE
      K=0
      DO 250 I=1,NM
      IF(PI(I).NE.0) GO TO 250
      K=K+1
      L=0
      DO 25 J=1,NM
      IF(PI(J).NE.0) GO TO 25
      L=L+1
      IF(K.GT.L) GO TO 26
      A(K+3,L+3)=AS(I,J)
      GO TO 25
   26 A(K+3,L+3)=A(L+3,K+3)
   25 CONTINUE
  250 CONTINUE
C
C     SOLVE SYSTEM MATRIX FOR REFERENCE BODY ANG. ACCELERATION AND HINGE
C                   (RELATIVE) ROTATIONAL ACCELERATIONS
C
      NT=V+NTMO
      IT=IV+NTMO
      CALL AINVD(A,ST,IT,S1095,WRK)
 1095 CONTINUE
      RETURN
      ENTRY MRATE(NC,TM,TB,TA,FB,FA,TF,FF,GM,GMD,GMDD,ET,ETD,WO,WDOT,ETD
     SD,HM)
      REAL TF(QF,NK,3),FF(QF,NK,3),ET(QF,NKT),ETD(QF,NKT),TB(3),TA(NC,3)
     S,FB(3),FA(NC,3),GM(1),GMD(1),GMDD(1),TM(1),WO(3),E(S3,1)
      DOUBLE PRECISION EC(ST),ETDD(QF,NKT),WDOT(V),EQ(ST)
C
C     BODY-TO-BODY COORDINATE TRANSFORMATION MATRICES
C
      DO 335 J=1,NM
      MM=J-1
      N=M(J)+1
      AB(1,1)=1.
      AB(1,2)=GM(J)*G(J,3)
      AB(1,3)=-GM(J)*G(J,2)
      AB(2,1)=-AB(1,2)
```

```
      AB(2,2)=1.
      AB(2,3)=GM(J)*G(J,1)
      AB(3,1)=-AB(1,3)
      AB(3,2)=-AB(2,3)
      AB(3,3)=1.
      IF(J.EQ.1) GO TO 3350
      DO 321 L=MM,1
      IF(EPS(L,N).EQ.1) GO TO 322
  321 CONTINUE
      GO TO 3350
  322 K=L
      DO 334 L=1,3
      DO 334 M=1,3
      T(J,L,M)=0.
      DO 334 I=1,3
  334 T(J,L,M)=T(J,L,M)+AB(L,I)*T(K,I,M)
      GO TO 335
 3350 CONTINUE
      DO 3351 L=1,3
      DO 3351 M=1,3
 3351 T(J,L,M)=AB(L,M)
  335 CONTINUE
C
C     COORD. TRANSFORMATION OF G VECTORS (TO REF. BODY FRAME)
C
      DO 362 I=1,NH
      DO 362 J=1,3
      GO(I,J)=0.
      DO 362 K=1,3
      GO(I,J)=GO(I,J)+T(I,K,J)*G(I,K)
  362 CONTINUE
C
C     COMPUTE TOTAL EXTERNAL FORCE ON EACH SUBSTRUCTURE (IN REF. COORD.)
C
      FEXO(1)=FB(1)
      FEYO(1)=FB(2)
      FEZO(1)=FB(3)
      IF(FI(1).EQ.0) GO TO 254
      IL=FI(1)
      JN=F(IL,2)
      DO 253 J=1,JN
      FEXO(1)=FEXO(1)+FF(IL,J,1)
      FEYO(1)=FEYO(1)+FF(IL,J,2)
  253 FEZO(1)=FEZO(1)+FF(IL,J,3)
  254 CONTINUE
      FS(1,1)=FEXO(1)
      FS(1,2)=FEYO(1)
      FS(1,3)=FEZO(1)
      DO 246 N=2,NB
      K=N-1
      DO 2460 L=1,3
 2460 FS(N,L)=FA(K,L)
      IF(FI(N).EQ.0) GO TO 246
      IL=FI(N)
      JN=F(IL,2)
      DO 245 J=1,JN
      DO 245 I=1,3
  245 FS(N,I)=FS(N,I)+FF(IL,J,I)
  246 CONTINUE
C
C     COMPUTE TRANSL. AND ROTAT. DISPLACEMENTS OF APPENDAGE SUB-BODIES
C
      DO 232 K=1,NF
      JN=F(K,2)
      LK=F(K,3)
      DO 233 J=1,JN
      DO 233 I=1,3
      U(K,J,I)=0.
      ID=(J-1)*6+I
```

```
      DO 233 L=1,LK
  233 U(K,J,I)=U(K,J,I)+EIG(K,ID,L)*ET(K,L)
  232 CONTINUE
C
C     COMPUTE C.M. PERTURBATION (FROM NOM. UNDEFORMED LOCATION) ON EACH
C                SUBSTRUCTURE WITH AN APPENDAGE (LOCAL COORDS.)
C
      DO 262 K=1,NF
      IK=F(K,1)+1
      JN=F(K,3)
      DO 263 I=1,3
  263 MCK(K,I)=0.
      DO 265 J=1,JN
      DO 265 I=1,3
  265 MCK(K,I)=MCK(K,I)-PK(K,I,J)*ET(K,J)
      DO 266 I=1,3
  266 CK(K,I)=MCK(K,I)/MSB(IK)
  262 CONTINUE
C
C     COMPUTE TOTAL EXTERNAL TORQUE ON EACH SUBSTRUCTURE W.R.T. ITS
C                INSTANTANEOUS C.M. (IN LOCAL COORD.)
C
      DO 268 L=1,3
  268 TS(1,L)=TB(L)
      DO 267 N=2,NB
      K=N-1
      DO 267 L=1,3
  267 TS(N,L)=TA(K,L)
      DO 2670 N=1,NB
      IL=FI(N)
      IF(IL.EQ.0) GO TO 2670
      JN=F(IL,2)
      DO 2671 J=1,JN
      DO 2671 L=1,3
 2671 TS(N,L)=TS(N,L)+TF(IL,J,L)
 2670 CONTINUE
      DO 269 N=1,NB
      K=FI(N)
      IF(K.EQ.0) GO TO 269
      TS(N,1)=TS(N,1)+CK(K,2)*FS(N,3)-CK(K,3)*FS(N,2)
      TS(N,2)=TS(N,2)+CK(K,3)*FS(N,1)-CK(K,1)*FS(N,3)
      TS(N,3)=TS(N,3)+CK(K,1)*FS(N,2)-CK(K,2)*FS(N,1)
  269 CONTINUE
      DO 271 N=1,NB
      K=FI(N)
      IF(K.EQ.0) GO TO 271
      JN=F(K,2)
      DO 272 J=1,JN
      RUX=RF(K,J,1)+U(K,J,1)
      RUY=RF(K,J,2)+U(K,J,2)
      RUZ=RF(K,J,3)+U(K,J,3)
      TS(N,1)=TS(N,1)+RUY*FF(K,J,3)-RUZ*FF(K,J,2)
      TS(N,2)=TS(N,2)+RUZ*FF(K,J,1)-RUX*FF(K,J,3)
  272 TS(N,3)=TS(N,3)+RUX*FF(K,J,2)-RUY*FF(K,J,1)
  271 CONTINUE
C
C     TRANSFORM VECTORS TO REF. BODY FRAME
C
      TXO(1)=TS(1,1)
      TYO(1)=TS(1,2)
      TZO(1)=TS(1,3)
      DO 17 I=2,NB
      M=HI(I)
      K=I-1
      L=C(K,1)+1
      FEXO(I)=T(M,1,1)*FS(I,1)+T(M,2,1)*FS(I,2)+T(M,3,1)*FS(I,3)
      FEYO(I)=T(M,1,2)*FS(I,1)+T(M,2,2)*FS(I,2)+T(M,3,2)*FS(I,3)
      FEZO(I)=T(M,1,3)*FS(I,1)+T(M,2,3)*FS(I,2)+T(M,3,3)*FS(I,3)
      TXO(I) =T(M,1,1)*TS(I,1)+T(M,2,1)*TS(I,2)+T(M,3,1)*TS(I,3)
```

```
      TYO(I) =T(M,1,2)*TS(I,1)+T(M,2,2)*TS(I,2)+T(M,3,2)*TS(I,3)
      TZO(I) =T(M,1,3)*TS(I,1)+T(M,2,3)*TS(I,2)+T(M,3,3)*TS(I,3)
      DXO(I,I)=T(M,1,1)*DX(I,I)+T(M,2,1)*DY(I,I)+T(M,3,1)*DZ(I,I)
      DYO(I,I)=T(M,1,2)*DX(I,I)+T(M,2,2)*DY(I,I)+T(M,3,2)*DZ(I,I)
      DZO(I,I)=T(M,1,3)*DX(I,I)+T(M,2,3)*DY(I,I)+T(M,3,3)*DZ(I,I)
      DXO(I,L)=T(M,1,1)*DX(I,L)+T(M,2,1)*DY(I,L)+T(M,3,1)*DZ(I,L)
      DYO(I,L)=T(M,1,2)*DX(I,L)+T(M,2,2)*DY(I,L)+T(M,3,2)*DZ(I,L)
      DZO(I,L)=T(M,1,3)*DX(I,L)+T(M,2,3)*DY(I,L)+T(M,3,3)*DZ(I,L)
      DO 17 J=1,NB
      IF(I.EQ.J) GO TO 17
      IF(CPS(K,J).EQ.1) GO TO 177
      IF(C(K,1).EQ.(J-1)) GO TO 17
      DXO(I,J)=DXO(I,L)
      DYO(I,J)=DYO(I,L)
      DZO(I,J)=DZO(I,L)
      GO TO 17
177   DXO(I,J)=T(M,1,1)*DX(I,J)+T(M,2,1)*DY(I,J)+T(M,3,1)*DZ(I,J)
      DYO(I,J)=T(M,1,2)*DX(I,J)+T(M,2,2)*DY(I,J)+T(M,3,2)*DZ(I,J)
      DZO(I,J)=T(M,1,3)*DX(I,J)+T(M,2,3)*DY(I,J)+T(M,3,3)*DZ(I,J)
17    CONTINUE
      DO 367 I=1,NB
      DXO(I,I)=DX(I,I)
      DYO(I,I)=DY(I,I)
367   DZO(I,I)=DZ(I,I)
C
C     COMPUTE TOTAL EXTERNAL FORCE ON VEHICLE (IN REF. COORD.)
C
      FTXO=0.
      FTYO=0.
      FTZO=0.
      DO 247 N=1,NB
      FTXO=FTXO+FEXO(N)
      FTYO=FTYO+FEYO(N)
247   FTZO=FTZO+FEZO(N)
C
C     COMPUTE THE PGSO VECTORS FOR EACH FLEX. APPENDAGE
C
      DO 208 K=1,NF
      KK=F(K,1)+1
      M=MI(KK)
      JNT=F(K,3)
      IF(KK.EQ.1) GO TO 2090
      DO 209 I=1,3
      PGSO(K,I)=0.
      DO 209 J=1,3
209   PGSO(K,I)=PGSO(K,I)+T(M,J,I)*(-MCK(K,J))
      GO TO 208
2090  CONTINUE
      DO 2091 I=1,3
2091  PGSO(K,I)=-MCK(K,I)
208   CONTINUE
C
C     VECTOR CROSS PRODUCTS DESCRIBING SYSTEM ROTATIONAL COUPLING.
C       (QUADRATIC TERMS INVOLVING THE CONNECTING BODY ANGULAR
C       VELOCITIES AND THE MUTUAL BARYCENTER-HINGE VECTORS)
C
      DO 230 N=1,NB
      I=FI(N)
      CPX=0.
      CPY=0.
      CPZ=0.
      DO 2301 L=1,NB
      CPX=CPX+DYO(N,L)*FEZO(L)-DZO(N,L)*FEYO(L)
      CPY=CPY+DZO(N,L)*FEXO(L)-DXO(N,L)*FEZO(L)
      CPZ=CPZ+DXO(N,L)*FEYO(L)-DYO(N,L)*FEXO(L)
2301  CONTINUE
      IF(I.EQ.0) GO TO 243
      FACT=MSB(N)/TM
      FTXM=FTXO*FACT
```

```
      FTYM=FTYO*FACT
      FTZM=FTZO*FACT
      PGFX=(PGSO(I,2)*(FEZO(N)-FTZM)-PGSO(I,3)*(FEYO(N)-FTYM))/MSB(N)
      PGFY=(PGSO(I,3)*(FEXO(N)-FTXM)-PGSO(I,1)*(FEZO(N)-FTZM))/MSB(N)
      PGFZ=(PGSO(I,1)*(FEYO(N)-FTYM)-PGSO(I,2)*(FEXO(N)-FTXM))/MSB(N)
      GO TO 244
243   CONTINUE
      PGFX=0.
      PGFY=0.
      PGFZ=0.
244   CONTINUE
      K = 3*(N-1)
      E(K+1,1)=TXO(N)+CPX+PGFX
      E(K+2,1)=TYO(N)+CPY+PGFY
      E(K+3,1)=TZO(N)+CPZ+PGFZ
230   CONTINUE
C
C     COMPUTE CORRECTION ELEMENTS FOR (E) VECTOR
C
      DO 55 MI=1,3
55    EC(MI)=E(MI,1)
      DO 52 J=2,NB
      DO 52 M=1,3
      K1=3*(J-1)+M
52    EC(M)=EC(M)+E(K1,1)
      I=0
      DO 60 K=1,NH
      JK=H(K)+1
      IF(PI(K).NE.0) GO TO 60
      I=I+1
      EC(I+3)=0.
      DO 601 M=1,3
601   CE(M)=0.
      DO 61 J=JK,NB
      IF(EPS(K,J).EQ.0) GO TO 61
      DO 65 M=1,3
      J1=3*(J-1)+M
65    CE(M)=CE(M)+E(J1,1)
61    CONTINUE
      DO 66 L=1,3
66    EC(I+3)=EC(I+3)+GO(K,L)*CE(L)
      EC(I+3)=EC(I+3)+TH(K)
60    CONTINUE
      DO 610 I=1,3
      DO 610 J=1,NH
      IF(PI(J).EQ.0) GO TO 610
      EC(I)=EC(I)-AV(J,I)*GMDD(J)
610   CONTINUE
      K=0
      IV=3
      DO 612 I=1,NH
      IF(PI(I).NE.0) GO TO 612
      K=K+1
      IV=IV+1
      DO 611 J=1,NH
      IF(PI(J).EQ.0) GO TO 611
      IF(I.GT.J) AS(I,J)=AS(J,I)
      EC(K+3)=EC(K+3)-AS(I,J)*GMDD(J)
611   CONTINUE
612   CONTINUE
C
C     COMPUTE RT. HAND SIDE OF APPENDAGE EQUATIONS (IN APPEND. COORDS.)
C
      DO 483 K=1,NF
      I=F(K,1)+1
      M=HI(I)
      CQ(1)=FTXO/TM
      CQ(2)=FTYO/TM
      CQ(3)=FTZO/TM
      IF(I.EQ.1) GO TO 484
```

```
        DO 484 J=1,3
        VE(K,J)=0.
        DO 484 L=1,3
  484   VE(K,J)=VE(K,J)+T(M,J,L)*CQ(L)
        GO TO 483
  4840  CONTINUE
        DO 4841 J=1,3
  4841  VE(K,J)=C4(J)
  483   CONTINUE
        DO 485 K=1,NF
        NL=F(K,2)
        DO 486 N=1,NL
        N6=6*(N-1)
        DO 488 J=1,3
        JN=N6+J
        JM=JN+3
        VB(K,JN)=FF(K,N,J)
  488   VB(K,JM)=TF(K,N,J)
  486   CONTINUE
  485   CONTINUE
        NV=IV
        DO 491 K=1,NF
        JN=F(K,3)
        NL=F(K,2)
        NL6=6*NL
        DO 492 J=1,JN
        IL=NV+J
        VV1=-WF(K,J)*(2.*ZF(K,J)*ETD(K,J)+WF(K,J)*ET(K,J))
        DO 493 N=1,NL6
  493   VV1=VV1+EIG(K,N,J)*VB(K,N)
        DO 494 N=1,3
  494   VV1=VV1-PK(K,N,J)*VE(K,N)
        EC(IL)=VV1
        DO 4920 L=1,NH
        IF(PI(L).EQ.0) GO TO 4920
        EC(IL)=EC(IL)-AKF(L,K,J)*GMDD(L)
  4920  CONTINUE
  492   CONTINUE
  491   NV=NV+JN
C
C       ANGULAR MOMENTUM OF THE SYSTEM
C
        IF(PI(NH+1).NE.1) GO TO 8752
        DO 5651 I=1,3
        HH(I)=0.
        DO 5651 J=1,3
  5651  HH(I)=HH(I)+AOD(I,J)*WO(J)
        DO 5652 I=1,3
        DO 5652 J=1,QH
  5652  HH(I)=HH(I)+AV(J,I)*GMD(J)
        DO 5653 I=1,3
        DO 5653 K=1,NF
        NL=F(K,3)
        DO 5654 J=1,NL
  5654  HH(I)=HH(I)+AOF(K,I,J)*ETD(K,J)
  5653  CONTINUE
        HM=SQRT(HH(1)**2 + HH(2)**2 + HH(3)**2)
  8752  CONTINUE
C
C       SOLVE SYSTEM MATRIX FOR REF. BODY ANG. ACCEL., SUBSTRUCTURE
C                HINGE ANGLE ACCEL., AND FLEX. BODY MODE ACCEL.
C
        DO 671 I=1,IT
        EQ(I)=0.
        DO 671 J=1,IT
  671   EQ(I)=EQ(I)+A(I,J)*EC(J)
        KV=IV
        DO 910 J=NT,4,-1
        IF(J.LE.V) GO TO 913
```

```
      JV=J-(V-IV)
      EC(J)=EQ(JV)
      GO TO 910
  913 CONTINUE
      K=J-3
      IF(PI(K).NE.0) GO TO 911
      EC(J)=EQ(KV)
      KV=KV-1
      GO TO 910
  911 EC(J)=GMDD(K)
  910 CONTINUE
      DO 6710 I=1,3
 6710 EC(I)=EQ(I)
      DO 9003 I=1,V
 9003 WDOT(I)=EC(I)
      I=V
      DO 9001 K=1,NF
      NL=F(K,3)
      DO 9002 N=1,NL
      IO=I+N
 9002 ETDD(K,N)=EC(IO)
 9001 I=I+NL
   92 CONTINUE
      RETURN
      END

DIAGNOSTICS

ATION TIME =   24.48 SUPS


CSSL.TRAN,CSSL
```

ORIGINAL PAGE IS OF POOR QUALITY

NASA—JPL—Coml., L.A., Calif.