Symbolic Vector/Dyadic Multibody Formalism for Tree-Topology Systems

Michael W. Sayers
University of Michigan Transportation Research Institute, Ann Arbor, Michigan 48109

A multibody formalism is presented that can be applied to automatically generate efficient equations of motion for a system of rigid bodies in a tree topology. The formalism is built on Kane's analysis method and is described using vector/dyadic notation. In addition to defining a way to formulate equations of motion, it specifies many details of the analysis that have formerly involved judgments made by a dynamicist. These details include "rule of thumb" issues such as 1) making modeling simplifications, 2) choosing state variables, 3) introducing intermediate variables, 4) choosing coordinate systems to represent vectors, and 5) choosing recursive vs nonrecursive formulations. The formalism has been automated using a computer algebra language that supports vector/dyadic algebra, small variable simplification options, and the automated introduction of new symbols. A companion paper describes this language and provides details of an example of a spacecraft multibody system. Results shown in this paper for the example spacecraft illustrate the high computational efficiency of the simulation code.

Introduction

THERE is a large body of literature covering systematic analysis methods for mechanical systems composed of rigid bodies. These methods are called multibody formalisms. The formalisms, all based on first principles, nonetheless represent a variety of analytical approaches. Large sets of simple dynamical and algebraic constraints equations can be formed into large matrices that are handled using sparse-matrix methods[1] or coordinate partitioning.[2] Minimal sets of dynamical equations involving Lagrangian generalized coordinates can be derived for tree-topology systems[3] and combined with algebraic constraint equations to handle kinematical loops.[4-6] Kane's method[7-10] and others are based on the principle that the virtual power of constraint forces and moments are identically zero.[4-6] Although most formalisms produce the equations of motion in implicit form, as a set of coupled differential equations, recursive, order(n) formalisms produce equations of motion in explicit form.[11]

In all of these formalisms, the equations of motion are essentially defined for once and for all in generic form before the analysis starts, and the analysis consists of plugging in coordinates and inertia parameters in the appropriate spots in the equations. When programmed, they require as input only a description of the layout of the multibody system. That description, followed by the specified manipulations, results in equations of motion that can be solved by computer. These formalisms include all coordinate system transformations and require only simple scalar arithmetic operations for their implementation. All could conceivably be implemented either numerically or symbolically. This approach contrasts with the strategies and tactics usually employed by a human dynamicist deriving equations by hand. For example, a dynamicist uses knowledge of the specific system to guide the analysis, performing extensive simplifications from the very start when possible.

The purpose of this paper is to describe a formalism that follows the approach taken by a human analyst. Rather than developing a set of generic equations for all systems, a rule-based procedure is developed for formulating equations using the approach advocated by Kane and Levinson.[7,8] Kane's method has been programmed in prior work,[9,10] but not with the generality needed to include arbitrary forces and moments acting on the system. To obtain the desired generality, equations are presented in basis-free vector notation. This contrasts with past work in which vector bases are determined once and for all by the author of the formalism. In fact, much of the complexity of the formalisms based on tree topologies has involved carefully specified matrix operations that provide required coordinate transformations.[3-6,11] The basis-free vector/dyadic representation used in this paper would be difficult to implement numerically but can be applied symbolically by a dynamicist using pencil and paper. More practically, the method is applied automatically by a computer program using symbolic computation methods that are described in a companion paper.[12]

This paper is limited in scope to holonomic tree-topology systems composed of rigid bodies. Systems with nonholonomic constraints or closed kinematical loops are handled by

Dr. Sayers received B.S. and M.S. degrees in Mechanical Engineering from the Massachusetts Institute of Technology in 1975 and 1976, respectively, and a Ph.D. in Mechanical Engineering from the University of Michigan in 1990. Since 1976, he has been a researcher at the University of Michigan Transportation Research Institute, conducting research involving the dynamics of ground vehicles, vehicle-road interactions, computer-aided analysis, and the automated modeling of dynamic mechanical systems. He developed the International Roughness Index used worldwide by highway agencies to characterize road roughness, and has designed and authored numerous software packages, most recently with emphasis on symbolic computation methods.

extending the methods presented here. (The software used to demonstrate the methods does in fact have the ability to handle the more general topologies.[13])

Notational Conventions

The state variables for a multibody system include ν generalized coordinates (designated q_1, q_2, ..., q_n) and ν independent speeds (designated u_1, u_2, ..., u_ν). The system is said to have ν degrees of freedom (DOF). The dimensions n and ν differ when 1) coordinates that are not of interest are omitted or 2) the system is subject to nonholonomic constraints. However, within the context of this paper, $n = \nu$. The equations of motion are ordinary differential equations that are commonly classified into two groups: kinematical and dynamical. The kinematical equations are used to compute derivatives of the generalized coordinates. In matrix form, they are

$$\underline{S}\underline{\dot{q}} = \underline{v} \tag{1}$$

where $\underline{S}$ is an $n \times n$ matrix, $\underline{\dot{q}}$ is a column array of length n containing the derivatives of the generalized coordinates, and $\underline{v}$ is a column array of length n.

The dynamical equations are used to compute derivatives of the independent speeds (accelerations). In matrix form, they are

$$\underline{M}\underline{\dot{u}} = \underline{f} \tag{2}$$

where $\underline{M}$ is a $\nu \times \nu$ matrix called the mass matrix, $\underline{\dot{u}}$ is a column array containing the ν derivatives of independent speeds, and $\underline{f}$ is a column array of length ν, called the force array.

The first objective of the symbolic analysis is to obtain the elements of the arrays $\underline{S}$, $\underline{v}$, $\underline{M}$, and $\underline{f}$. The equations can be coded in a computer language to define a procedure that calculates values for the derivatives of the coordinates and speeds, given values of the coordinates, the speeds, and time. The procedure that computes derivatives is part of a computer program called a simulation code that numerically integrates the differential equations to "simulate" the multibody system. The algorithm that computes the derivatives is the actual end product of the analysis. Because the calculations of the derivatives are repeated many times in most simulation codes, the efficiency of the algorithm is of great interest when computation time must be minimized.

Bodies in the multibody system are designated by plain capital letters, e.g., body A, body B. The inertial reference is called N. Points are designated by capital letters that often have subscripts. Origins of coordinate systems are always written with a subscript zero (e.g., B_0). When discussing bodies in the system, the current (generic) body under consideration is called B. Its movements are defined with reference to another body in the system, called the parent of B, and designated A. (The parent A can be either another body or N.) The configuration of the multibody system when all generalized coordinates are zero is called the nominal configuration.

Vectors are written with bold type. Unit vectors that are parallel with axes in coordinate systems are written with a lower-case letter that is the same as the body in which the unit vector is fixed, and subscripted with an index of 1, 2, or 3. For example, the three directions of the coordinate system of B are the unit vectors b_1, b_2, and b_3. Other unit vectors, used to define directions of interest, are written with the letter d. Position vectors are written with the letter r, superscripted with the names of the endpoints of the vector. For example, a vector connecting the origin of $B(B_0)$ to its mass center (B^*) is $r^{B_0B^*}$. The absolute velocity of a point is written with the letter v superscripted with the symbol for the point. For example, the absolute velocity of point B^* is v^{B^*}. An incremental velocity is defined as the difference between the absolute velocities of two points, and is written with the two points in the superscript, e.g.,

$$v^{A^*B^*} \equiv v^{B^*} - v^{A^*} \tag{3}$$

Acceleration and incremental acceleration vectors are written in the same fashion used for velocity, but using the letter a (e.g., a^{B^*}, $a^{A^*B^*}$). Angular velocity is written with the symbol ω, e.g., ω^B. Incremental angular velocity is the same as relative angular velocity. The angular velocity of B relative to A is written $^A\omega^B$. Angular acceleration is written using the letter α, e.g., α^B. Incremental angular acceleration is written as

$$\alpha^{AB} \equiv \alpha^B - \alpha^A \tag{4}$$

Incremental velocity, incremental acceleration, and incremental angular acceleration are not the same as relative velocity, relative acceleration, and relative angular acceleration. (However, incremental angular velocity is the same as relative angular velocity.)

A given vector can be written many ways, using alternative unit vectors. A vector written using only the three unit vectors aligned along the axes of the coordinate system of body B is said to be expressed in the basis of B. A vector written with no explicit trigonometric functions is said to be expressed in native form. For example, consider a system of two bodies A and B, where $a_1 = n_1$ and $b_2 = a_2$, and the angular velocity of B, ω^B, expressed in native form, is

$$\omega^B = u_1 n_1 + u_2 a_2 \tag{5}$$

Next, consider a dyadic obtained by "doubling" three mutually orthogonal unit vectors and adding the results. This type of dyadic is called a basis dyadic. For example, three such dyadics are

$$\vec{\vec{N}} = n_1 n_1 + n_2 n_2 + n_3 n_3$$

$$\vec{\vec{A}} = n_1 n_1 + a_2 a_2 + a_3 a_3$$

$$\vec{\vec{B}} = b_1 b_1 + a_2 a_2 + b_3 b_3 \tag{6}$$

Dotting a vector with a basis dyadic projects the vector into the basis associated with the dyadic. However, the new vector has the same magnitude and direction as the original vector:

$$\omega^B = \omega^B \cdot \vec{\vec{N}} = \omega^B \cdot \vec{\vec{A}} = \omega^B \cdot \vec{\vec{B}} \tag{7}$$

The written representation of the vector may be changed by the projection operation. For example, the three dot products shown in Eq. (7) are written as follows:

$$\omega^B = \omega^B \cdot \vec{\vec{N}} = u_1 n_1 + u_2 C_1 n_2 + u_2 S_1 n_3$$

$$= \omega^B \cdot \vec{\vec{A}} = u_1 n_1 + u_2 a_2$$

$$= \omega^B \cdot \vec{\vec{B}} = u_1 C_2 b_1 + u_2 a_2 + u_1 S_2 B_3 \tag{8}$$

where C_1, S_1, C_2, and S_2 are cosine and sine functions associated with joint rotation angles. When the equations of motion are programmed for numerical solution, efficiency is improved by ensuring that each arithmetic operation between two variables is performed only once. The first occurrence of an expression that appears more than once is replaced with an intermediate variable, and the intermediate variable is used subsequently. The replacement of an expression with an intermediate variable is indicated in this paper by enclosing the expression with the symbols "«" and "»". For example, the expression «$\omega^B \cdot \vec{\vec{B}}$» is interpreted as: "take the dot product as indicated, then replace scalar expressions with intermediate variables." For the previous example, the result would be an expression similar to the following:

$$\text{«}\omega^B \cdot \vec{\vec{B}}\text{»} \rightarrow Z_8 b_1 + u_2 a_2 + Z_9 b_3 \tag{9}$$

where Z_8 and Z_9 are intermediate variables introduced for the expressions $u_1 C_2$ and $u_1 S_2$, respectively. All expressions devel-

oped later involving «$\omega^B \cdot \vec{B}$» would include Z_8 and Z_9 rather than $u_1 C_2$ and $u_1 S_2$. (Details of how an expression is processed to define intermediate variables are provided in the companion paper.[12])

The choice of a vector basis and the use of intermediate variables are of no consequence with respect to the correctness of the equations of motion. However, these choices can strongly influence the efficiency of the resulting simulation code. Accordingly, they are considered in the formalism that will be presented.

Dynamics Analysis via Kane's Method

Starting with knowledge of the multibody system of interest, and desiring equations of the form of Eqs. (1) and (2) as the final product, an analysis based on Kane's method might proceed as follows. First, the analyst decides how many bodies are used to represent the system and how they are related to each other kinematically by joints. For each body, three mutually orthogonal unit vectors are established to define directions and positions relative to that body. All of the force and torque-producing components are identified, including moments caused by forces whose lines of action do not pass through the mass centers of the bodies upon which they act. Generalized coordinates and generalized speeds are introduced as state variables, such that it is possible to write an expression for the instantaneous position and velocity of any point on any body in the system using dimensional parameters and the state variables.

As the state variables are introduced, the analyst formulates kinematical equations that define the derivatives of the generalized coordinates in terms of the generalized speeds [see Eq. (1)]. Often, the speeds are simply the derivatives of the coordinates (i.e., $\dot{q}_i = u_i$). However, the analyst is permitted to introduce speeds that are not simply derivatives of the generalized coordinates, sometimes called derivatives of quasi-coordinates.

The analyst develops an expression for the angular velocity vector of each body B, ω^B. From each angular velocity, ν terms called partial angular velocities are defined:

$$\omega_i^B = \frac{\partial \omega^B}{\partial u_i} \quad (i = 1, \ldots, \nu) \tag{10}$$

A partial angular velocity is simply a coefficient appearing in an expression for angular velocity. Because angular velocity is a vector, and speed is scalar, it follows that a partial angular velocity is always a vector. The total number of partial angular velocities that exists for the multibody system is the product νN_B, where N_B is the number of rigid bodies in the system.

Next, expressions are developed for the velocity vectors of the mass centers of each body, v^{B*}. From these expressions, ν terms called partial velocities are defined:

$$v_i^{B*} = \frac{\partial v^{B*}}{\partial u_i} \quad (i = 1, \ldots, \nu) \tag{11}$$

The ν dynamical equations for the system can be written as follows, for $i = 1, \ldots, \nu$:

$$0 = \sum_B^{N_B} \left[\left(\sum_{t=1}^{N_{B,T}} T_t^B - \alpha^B \cdot \overleftrightarrow{I}^{B*} - \omega^B \times \overleftrightarrow{I}^{B*} \cdot \omega^B \right) \cdot \omega_i^B \right. \\ \left. + \left(\sum_{f=1}^{N_{B,F}} F_f^B - m^B a^{B*} \right) \cdot v_i^{B*} \right] \tag{12}$$

In the previous equation set, the outer summation, with index B, is meant to imply summing over all bodies in the system. The number of torques acting on body B is designated $N_{B,T}$, and the individual torques are designated T_t^B. Similarly, the number of forces acting on body B is designated $N_{B,F}$, and the individual forces are designated F_f^B. Note that forces acting on B can appear twice: once as a force F_f^B, and once as a coefficient in a moment about the mass center of $B*$,

$$T = r^{B*P} \times F \tag{13}$$

where P is a point in B on the line of action of the force vector F. The mass of B is designated m^B, and the inertia dyadic of B (with respect to its mass center) is designated $\overleftrightarrow{I}^{B*}$.

The analyst can convert the dynamical equations from the form of Eq. (12) into the desired form of Eq. (2) by inspection, moving terms containing the speed derivatives to the left side.

Multibody Formalism

The remainder of this paper describes a formal process (i.e., a multibody formalism) by which an algorithm for computing values of the derivatives of the state variables is developed from a geometric description of a multibody system. The process is performed in three consecutive steps: 1) system description, 2) kinematics analysis, and 3) dynamics analysis.

In developing the formalism, we assume a computer language exists that is capable of manipulating numbers, scalar algebraic expressions, vectors, dyadics, and points. We also assume that a function exists to introduce a new intermediate variable for scalar expressions contained within vectors, as illustrated in Eq. (9). The formalism presumes that bodies are represented in such a form that data associated with each body are easily accessible. It also presumes that vector dot-products and cross-products are formed automatically, based on the properties of the body connections. By assuming the existence of this specialized symbolic computer language, we can ignore many of the algebraic details that are handled symbolically and can concentrate on dynamics and kinematics. (The specialized symbolic computer language exists and is described in the companion paper.)

An example multibody system will be used to illustrate the methods that follow. The system, shown in Fig. 1, is a satellite with a main body B (called the bus), a flexible boom F, and a camera D, mounted on a clock C. Dimensions and locations of significant points are shown in Fig. 2. Further information about this system is provided in the companion paper.

Step 1: System Description

In the first step of the analysis, all of the bodies, joints, and force- and torque-producing elements are described by the
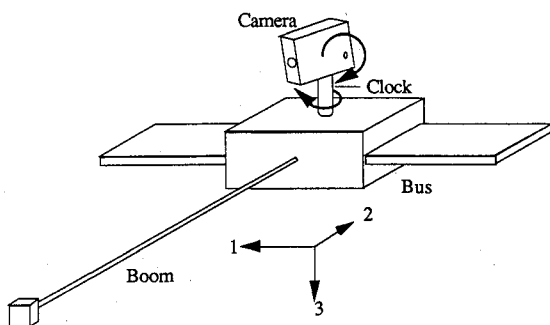

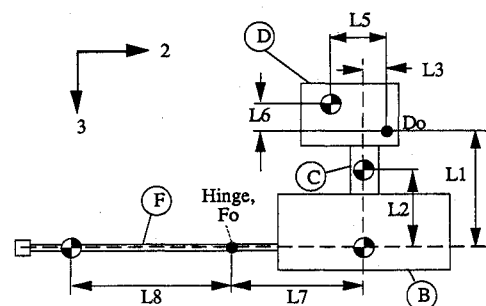Fig. 1 Example multibody system.

Fig. 2 Dimensions of satellite.

dynamicist. (The companion paper lists the description for the example system in a computer program that uses this formalism.) From that description, a set of expressions is developed to represent each element. Each rigid body in the system has associated with it a reference frame and a coordinate system. The analyses performed when a body is added deal mainly with the coordinate system of the new body, as determined by the kinematics of the joint connecting it to a parent body that has already been described. (The inertial reference N can also be specified as a parent.) Note that the convention of defining each new body as a child of an existing body naturally organizes the system into a tree topology.

Joint Description

A building-block joint model is used to define the kinematical relation between a new body and its parent. The joint includes between zero and six kinematical degrees of freedom. Three of these are consecutive translations, and the other three are consecutive simple rotations. The parameters that describe the building-block joint are summarized in Table 1, with optional items enclosed in curly braces. The geometry of a joint is illustrated in Fig. 3 for an example involving one degree of freedom for rotation and one for translation.

The relationship between the coordinate systems of B and A depends on the type and number of DOF:

1) If the joint has no translational degrees of freedom, B_0 is coincident with a point in A called the joint point and

designated B_J. Otherwise, it can move relative to B_J, in directions that are fixed in A.

2) If the joint has no rotational degrees of freedom, the axes in B may be parallel to those in A. Otherwise, the directions of at least two of the axes in B differ from those of A. (The dynamicist is permitted to change the orientation of B relative to A without including a rotational degree of freedom, to define a new coordinate system of interest.)

One generalized coordinate is introduced for each degree of freedom of the joint. In Fig. 3, the magnitude of the translation is the generalized coordinate q_i, and the magnitude of the rotation is the generalized coordinate q_{i+1}.

Table 2 shows how the joints of the satellite example are characterized. The first body B has six joint degrees of freedom: three in translation and three in rotation. The other bodies each have just one joint degree of freedom in rotation.

The number of translational degrees of freedom N_{td}^B is determined by the length of the list of their directions. The position of point B_0 relative to point A_0 is the vector

$$r^{A_0B_0} = r^{A_0B_J} + \sum_{i=1}^{N_{td}^B} q_{i+o}d_{ti}^B \qquad (14)$$

where o is an offset constant that maps the index i from Eq. (14) into the indices used for the generalized coordinates. For the satellite, N_{td}^B is 3 for body B and zero for the other bodies.
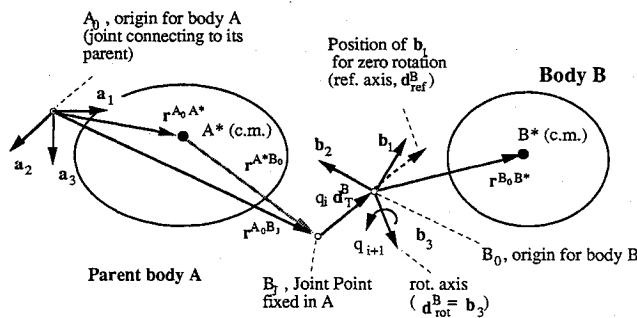

Fig. 3   Geometry of body relative to its parent.

Table 1   Parameters and degrees of freedom of a body/joint

r^{A_0BJ}	Position of joint point of B relative to origin of parent
$(\{d_{t1}^B, \{d_{t2}^B, \{d_{t3}^B\}\}\})$	List of 0, 1, 2, or 3 directions for translational degrees of freedom of B, fixed in the coordinate system of the parent
$(\{i_1, \{i_2, i_3\}\})$	List of 0, 1, or 3 axis indices in B for sequential rotations
d_{rot}^B	Orientation of first rotation axis of B (fixed in the coordinate system of the parent)
d_{ref}^B	Reference direction for first rotation of B (fixed in the coordinate system of the parent)
$(\{d_{r1}^B, \{d_{r2}^B, \{d_{r3}^B\}\}\})$	List of 0, 1, or 3 directions of rotations for B (This list is derived from the above parameters.)

Table 2   Body and joint properties of satellite example[a]

Property	Body B	Body C	Body D	Body E	Body F
Name	Bus	Clock	Camera	E	Boom
Parent	N	B	C	B	E
Level	1	2	3	2	3
Children	C, E	D	—	F	—
Translation category	Nonrecursive	Fixed	Recursive	Fixed	Recursive
Composite body mass: M^{Bc}	$M_B + M_C$	M_C	M_D	0	M_F
Unit vectors	b_1, b_2, b_3	c_1, c_2, b_3	c_1, d_2, d_3	e_1, e_2, b_3	e_1, f_2, f_3
Rotation category	Nonrecursive	Rotor	Recursive	Rotor	Recursive
Inertia dyadic of composite body: $\overset{\leftrightarrow}{I}^{Bc*}$	$(I_{B11} + M_B P_1^2 + M_C(L_2 - P_1)^2$ $(b_1\ b_1) + I_{B12}(b_2\ b_1) + I_{B12}(b_1$ $b_2) + I_{B13}(b_1\ b_3) + (I_{B22} + M_B$ $P_1^2 + M_C(L_2 - P_1)^2)(b_2\ b_2) + I_{B13}$ $(b_3\ b_1) + I_{B23}(b_3\ b_2) + I_{B23}(b_2$ $b_3) + I_{B33}(b_3\ b_3)$	$I_C(b_1\ b_1) +$ $I_C(b_2\ b_2)$	$I_{D11}(c_1\ c_1) + I_{D12}(d_2$ $c_1) + I_{D12}(c_1\ d_2) +$ $I_{D13}(c_1\ d_3) + I_{D13}(d_3$ $c_1) + I_{D22}(d_2\ d_2) +$ $I_{D23}(d_3\ d_2) + I_{D23}(d_2$ $d_3) + I_{D33}(d_3\ d_3)$	0	$I_{F1}(e_1\ e_1) +$ $I_{F2}(f_2\ f_2) +$ $I_{F1}(f_3\ f_3)$
Translational coordinates	q_1, q_2, q_3				
Translational speeds	u_1, u_2, u_3				
Translational directions	n_1, n_2, n_3				
Direction of translational speeds	b_1, b_2, b_3				
Rotational coordinates	q_4, q_4, q_6	q_7	q_8	q_9	q_{10}
Rotation speeds	u_4, u_5, u_6	u_7	u_8	u_9	u_{10}
Rotation directions	$n_1, (C_6b_2 + S_6b_1), b_3$	b_3	c_1	b_3	e_1
Directions of rotational speeds	b_1, b_2, b_3	b_3	c_1	b_3	e_1
Joint point (in parent)	0, 0, 0	0, 0, 0	$0, L_3, -L_1$	$0, -L_7, 0$	0, 0, 0
Mass center of composite body	$0, 0, -P_1$	$0, 0, -L_2$	$0, -L_5, -L_6$	0, 0, 0	$0, -L_8, 0$

[a] $P_1 = L_2 M_C/(M_B + M_C)$.

The offset o is zero. In Fig. 3, N_{td}^B is 1 and the position vector $r^{A_0 B_0}$ is $r^{A_0 B_J} + q_i d_T^B$.

The number of rotational degrees of freedom is determined by the length of the list of axis indices. Two pieces of information are required in addition to the list to specify the orientation of B relative to A when all generalized coordinates are zero. First, the orientation of the first rotation axis d_{rot}^B is in a direction fixed in the parent. (The direction d_{rot}^B may or may not be parallel with an axis of the coordinate system of the parent.) Second, a reference direction d_{ref}^B defines the orientation of a reference axis in B. In the nominal configuration, the directions of three axes of the coordinate system of B are defined for three possible cases, based on the first rotation index i_1. Table 3 summarizes these three cases.

When the joint has three rotational degrees of freedom, the list of axis indices specifies the sequence of rotations. For the satellite example, a 1-2-3 rotation was used for the bus B. Table 2 shows that the three rotation angles are q_4, q_5, and q_6. The first rotation axis is n_1, the third is b_3, and the second, fixed in an intermediate frame, is $(C_6 b_2 + S_6 b_1)$, where C_6 and S_6 are the cosine and sine of q_6. There are six possible directions of the second rotation axis, corresponding to the possible combinations of the second and third rotation axes (i_2 and i_3). Table 4 lists these directions.

To simplify some of the rules that follow, the building-block joint model allows zero, one, or three consecutive rotations between a body and its parent, but not two rotations. Joints that involve two consecutive rotations are represented by two building-block joints, where the first is associated with a massless body. In the satellite model, the flexing of the boom in two directions is handled by introducing a massless body E with a rotational DOF. Together, bodies E and F describe a rigid body (with mass and inertia) connected to B by a joint that has two rotational degrees of freedom.

Inertia Properties of Bodies and Composite Bodies

As each body is entered, an analysis is performed to set the inertia properties of the new body and all bodies up the tree. This is done to make modeling simplifications when inertia properties from several bodies can be lumped together. For example, consider the clock (body C) in the satellite. Its mass center is located on the joint axis, and thus its mass can be lumped with the mass of the bus (body B). Also, the clock has products of inertia that are zero, and the same moment of inertia applies for any direction perpendicular to the spin axis (I_C). Thus, terms associated with the inertia dyadic of the clock can be expressed as constants when using directions fixed in the parent (see Table 2).

When each new body is added, the following procedure is applied to set inertia properties:

1) The body is placed into one of three categories based on the degrees of freedom of its joint and the location of its mass

center. The categories are defined in Table 5. A list, called the "fixed children of B" is made of all children of B that are in the category fixed. (The first time this procedure is applied to body B, there are no children and a null list is used.)

2) A composite mass for B is defined as the sum of the mass of B with the composite masses of the fixed children of B:

$$m^{Bc} = m^B + \sum_b^{\substack{\text{fixed} \\ \text{children}}} m^{bc} \qquad (15)$$

where m^{Bc} is the composite mass for B, and the sum covers the fixed children, with the index b indicating each body that is a member of the list of fixed children of B. For example, Table 2 shows a composite mass for the bus of $(M_B + M_C)$. The mass of a fixed body appears twice: once for the original body and once for the parent. Later, when equations of motion are formed, only the term from the parent is used.

3) The coordinates of the composite mass are computed:

$$x_i^{Bc*} = \frac{x_i^{B*} m^B + \sum_b^{\substack{\text{fixed} \\ \text{children}}} x_i^{bc*} m^{bc}}{m^{Bc}} \qquad (i = 1,2,3) \qquad (16)$$

where x_i^{B*} is one of the three coordinates of the mass center of B and x_i^{bc*} is the corresponding coordinate of the composite mass of a fixed child of B, which has been converted to the coordinate system of B. For example, Table 2 shows that the mass center for body B of the satellite has been modified to include the effect of fixed body C. (All other mass centers for the satellite are as shown in Fig. 2.)

4) An inertia matrix for the composite body is constructed using the parallel axes theorem, considering the masses of the fixed children:

$$I_{ii}^{Bc*} = I_{ii}^{B*} + \sum_b^{\substack{B, \text{ fixed} \\ \text{children}}} m^b[(x_j^{b*} - x_j^{Bc*})^2 + (x_k^{b*} - x_k^{Bc*})^2]$$

$$(i,j,k = 1,2,3; \; i \neq j \neq k)$$

$$I_{ij}^{Bc*} = I_{ji}^{Bc*} = I_{ij}^{B*} - \sum_b^{\substack{B, \text{ fixed} \\ \text{children}}} m^b[(x_i^{b*} - x_i^{Bc*})(x_j^{b*} - x_j^{Bc*})]$$

$$(i,j = 1,2,3; \; i \neq j) \qquad (17)$$

For example, in Table 2, the coefficients of the inertia dyadic of body B of the satellite include the effect of the fixed mass of body C. Inertia moments and products for the other bodies in the satellite are the unmodified inertia parameters of the rigid bodies.

5) The inertia matrix constructed in step 4 is made into a dyadic using the unit-vectors of B:

$$\overrightarrow{I}^{Bc*} = \sum_{i=1}^3 \sum_{j=1}^3 (I_{ij}^{Bc*} b_i b_j) \qquad (18)$$

6) The rotational category of B is determined according to criteria shown in Table 6. The inertia matrix from step 4 is

Table 3 Orientation of unit vectors in the nominal configuration

$i_1 = 1$	$i_2 = 2$	$i_3 = 3$
$b_1 = d_{rot}^B$	$b_2 = d_{rot}^B$	$b_3 = d_{rot}^B$
$b_3 = d_{rot}^B \times d_{ref}^B$	$b_1 = d_{rot}^B \times d_{ref}^B$	$b_2 = d_{rot}^B \times d_{ref}^B$
$b_2 = b_3 \times b_1$	$b_3 = b_1 \times b_2$	$b_1 = b_2 \times b_3$

Table 4 Direction of second rotation for joint with three rotations[a]

i_3	i_2	d_{r2}^B
1	2	$C_3 b_2 - S_3 b_3$
	3	$S_3 b_2 + C_3 b_3$
2	1	$C_3 b_1 + S b_3$
	3	$-S_3 b_1 + C_3 b_3$
3	1	$C_3 b_1 - S_3 b_2$
	2	$S_3 b_1 + C_3 b_2$

[a]S_3 and C_3 are the sine and cosine of the third rotation angle of the joint.

Table 5 Translation categories for a body

Category	Description	Criteria
Nonrecursive	v_i^{B*} can be described using only translational speed variables associated with B	Joint has 3 DOF in translation or [2 DOF and $\omega^B \times (d_{t1}^B \times d_{t2}^B) = 0$ and $v^{BJ} \cdot (d_{t1}^B \times d_{t2}^B) = 0$]
Fixed	Mass center is fixed in reference frame of parent	$^A v^{B*} = 0$
Recursive	Expression for v_i^{B*} involves velocity of A	Neither of the above two criteria are satisfied

Table 6 Rotation categories for a body

Category	Description	Criteria
Nonrecursive	Angular velocity of body can be described using only joint speeds associated with B	Joint has 3 DOF in rotation
Rotor	Inertia matrix can be expressed in coordinate system of parent body	0 rotational DOF or 1 rotational DOF and inertia matrix is diagonal with equal moments orthogonal to spin axis
Recursive	Expression for rotation of body involves rotation of its parent	1 rotational DOF and body is not a rotor

used to determine if B is a rotor. If it is, the inertia dyadic is converted to the basis of the parent using the identity

$$\vec{\vec{I}}^{Bc^*} = \vec{\vec{A}} \cdot \vec{\vec{I}}^{Bc^*} \cdot \vec{\vec{A}} \tag{19}$$

Otherwise, the formulation obtained from Eq. (18) is kept. For example, Table 2 shows that the inertia dyadic for body C of the satellite is expressed as in Eq. (19). All other inertia dyadics are expressed as in Eq. (18), except for the zero inertia of the massless intermediate body E.

7) The translational category of B is determined according to the criteria in Table 5.

8) The previous procedure is repeated for the parent of B, unless the parent is the inertial reference (N).

The last step in the previous procedure means that as each body is added to the tree, the mass and inertia properties of all bodies up the tree are subject to adjustment. For example, in the satellite, body B was first assigned inertial properties determined solely by the rigid-body characteristics of B alone. However, when body C was added and classified as fixed, the inertial properties of B were modified to include a point mass M_C located as shown in Fig. 2.

Speed Variables

One generalized speed is introduced for each joint degree of freedom. If the joint has translational degrees of freedom and is recursive or fixed in translation (see Table 5), the corresponding speeds are defined simply as the derivatives of the generalized coordinates. If the body is nonrecursive in translation, the corresponding speeds are defined as components of the velocity of the mass center of the composite body B, in directions parallel to the body fixed-axes.

If the joint has rotational degrees of freedom and is recursive or a rotor, the corresponding speeds are the derivatives of the generalized coordinates. If the body is nonrecursive in rotation, the corresponding speeds are defined as components of the angular velocity of B, in directions parallel to the body fixed axes.

For the satellite example, Table 2 shows that the speeds introduced for bodies C, D, E, and F are in the same directions as the corresponding coordinates. However, the speeds introduced for body B are defined for directions fixed in B (b_1, b_2, b_3), whereas the translational coordinates are defined in directions fixed in N, and the rotation axes are in three reference frames corresponding to sequential rotations of B.

Step 2: Kinematical Analysis

An analysis is performed to obtain the kinematical equations that define derivatives of the generalized coordinates in terms of known speeds, with the form of Eq. (1), $\underline{S}\underline{\dot{q}} = \underline{v}$.

Translational Coordinates

If the body is recursive in translation (see Table 5), the kinematical equation is simply

$$\dot{q}_i = u_i \quad \text{(recursive in translation)} \tag{20}$$

Otherwise, kinematical equations are obtained by considering the velocity of the body origin B_0 relative to the reference frame of A. This velocity can be written in terms of the derivatives of the generalized translational coordinates introduced for B:

$$^A v^{B_0} = \sum_{j=1}^{N_{td}^B} \dot{q}_{o+j} d_{tj}^B \tag{21}$$

An alternative expression can be written that involves the generalized speeds:

$$^A v^{B_0} = \sum_{j=1}^{N_{td}^B} u_{o+j} v_{o+j}^{B^*} - \omega^B \times r^{B_0 B^*} - v^{A_0}$$

$$\text{(nonrecursive in translation)} \tag{22}$$

In this equation, the direction associated with speed u_{o+j} is the corresponding partial velocity $v_{o+j}^{B^*}$, which is not always parallel with the direction associated with the translation d_{tj}^B.

A kinematical equation is obtained for $\dot{q}_i$ by equating the right sides of Eqs. (21) and (22), and dot-multiplying both by $d_{t,i-o}^B$:

$$\left(\sum_{j=1}^{N_{td}^B} \dot{q}_{o+j} d_{tj}^B \right) \cdot d_{t,i-o}^B$$

$$= \left(\sum_{j=1}^{N_{td}^B} u_{o+j} v_{o+j}^{B^*} - \omega^B \times r^{B_0 B^*} - v^{A_0} \right) \cdot d_{t,i-o}^B \tag{23}$$

Note that equations of the form of Eq. (23) reduce to the form of Eq. (20) when the body is recursive in translation and the translational directions are orthogonal. In terms of the matrix equations (1), the elements of S for row i are obtained from the left sides of Eq. (20) or Eq. (23), and the elements of $\underline{v}$ are obtained from the right side.

For the satellite example, the following kinematical equations are obtained for the translational coordinate derivatives:

$$P_1 = L_2 M_C / (M_B + M_C), \quad Z_1 = P_1 u_5 + u_1, \quad Z_2 = P_1 u_4 - u_2$$

$$q_1' = C_5 (Z_1 C_6 + Z_2 S_6) + u_3 S_5$$

$$q_2' = -[Z_2(C_6 C_4 - S_6 S_4 S_5) - Z_1(C_6 S_5 S_4 + C_4 S_6) + u_3 C_5 S_4]$$

$$q_3' = u_3 C_5 C_4 + Z_1(-C_6 C_4 S_5 + S_6 S_4) - Z_2(C_4 S_5 S_6 + C_6 S_4) \tag{24}$$

The symbol P_1 designates a constant that can be precomputed (it is a coordinate of a composite mass center and also appears in Table 2), the symbols Z_1 and Z_2 are intermediate variables, C_i is the cosine of q_i, S_i is the sine of q_i, and derivatives are indicated with a prime.

Rotational Coordinates

If the joint has one rotational degree of freedom, the kinematical equation is simply

$$\dot{q}_i = u_i \quad \text{(one rotational DOF)} \tag{25}$$

If the joint has three rotational DOF, we consider the three consecutive rotation angles of B relative to its parent A: q_{o+1}, q_{o+2}, and q_{o+3} (o is an index offset; for the satellite example, $o = 3$, and the first rotation is q_4). The angular velocity of the body, relative to its parent, can be written as follows:

$$^A \omega^B = \sum_{j=1}^{3} \dot{q}_{o+j} d_{rj}^B \tag{26}$$

where d_{rj}^B is the axis of rotation associated with q_{o+j}. The

angular velocity of B relative to A can also be expressed in terms of speed variables as follows:

$$^A\omega^B = \left(\sum_{j=1}^{3} u_{o+j} b_j\right) - \omega^A \tag{27}$$

A set of three kinematical equations is obtained by equating Eqs. (26) and (27) and dot-multiplying both sides by $\vec{d}_{rj}^{B}$ to yield the following for $i = o + 1$, $o + 2$, and $o + 3$:

$$\left(\sum_{j=1}^{3} \dot{q}_{o+j} d_{rj}^{B}\right) \cdot d_{r,i-o}^{B} = \left[\left(\sum_{j=1}^{3} u_{o+j} b_j\right) - \omega^A\right] \cdot d_{r,i-o}^{B}$$

(three rotational DOF) $\qquad$ (28)

Note that if Eq. (28) is applied for the case of one rotational DOF, it reduces to the form of Eq. (25).

For the satellite example, the following rotational kinematical equations are obtained:

$$q_4' = (u_4 C_6 - u_5 S_6)/C_5, \qquad q_5' = u_5 C_6 + u_4 S_6$$

$$q_6' = u_6 - q_4' S_5, \qquad q_7' = u_7$$

$$q_8' = u_8, \qquad q_9' = u_9, \qquad q_{10}' = u_{10} \tag{29}$$

Step 3: Dynamics Analysis

The unstructured form of the dynamical equations shown earlier in Eq. (12) requires further manipulation to obtain the matrix form commonly used for computer solution. Rather than deriving equations in the form of Eq. (12) and then converting them to the form of Eq. (2), we will generate them in the desired form.

Rotational Terms

The definition of a partial angular velocity [Eq. (5)] can be inverted to express the angular velocity in terms of the partial angular velocities:

$$\omega^B = \omega_t^B + \sum_{i=1}^{\nu} u_i \omega_i^B \tag{30}$$

where ω_t^B is an explicit function of time. In this paper, we restrict the analysis to systems in which ω_t^B is zero, such that

$$\omega^B = \sum_{i=1}^{\nu} u_i \omega_i^B \tag{31}$$

(This restriction is removed when nonholonomic constraints are included.[13])

Angular acceleration can also be expressed in terms of partial angular velocities:

$$\alpha^B = \frac{d\omega^B}{dt} = \sum_{r=1}^{\nu}\left(\frac{du_r}{dt}\omega_r^B + u_r\frac{d\omega_r^B}{dt}\right) = \sum_{r=1}^{\nu} \dot{u}_r \omega_r^B + \alpha_{rem}^B \tag{32}$$

where the angular acceleration remainder α_{rem}^B is defined as:

$$\alpha_{rem}^B = \sum_{i=1}^{\nu} u_r \frac{d\omega_i^B}{dt} \tag{33}$$

Comparing Eqs. (2) and (32), we see that the angular acceleration of B has been broken up into two parts: one that goes on the left side of the equations (in the mass matrix) and a remainder that goes on the right side (in the force array).

For each body in the system, we need to formulate expressions for the angular velocity ω^B, the angular acceleration remainder α_{rem}^B, and the ν partial angular velocities, $\omega_i^B (i = 1, \ldots, \nu)$. To start the analysis, the angular velocity and angular acceleration remainder for N are set to zero:

$$\alpha_{rem}^N = \omega^N = 0 \tag{34}$$

Also, the ν partial angular velocities for N are defined as zero:

$$\omega_i^N = 0 \quad (i = 1, \ldots, \nu) \tag{35}$$

The analysis proceeds such that children bodies are always processed after their parent. That is, the analysis proceeds down the topology tree. Table 7 summarizes the relationships used to develop the necessary expressions for every other body in the system based on how it is classified in rotation (see Table 5). Note that the formulas in the table must be applied in the order they appear (from the top to the bottom row).

The formulas in the table specify when intermediate variables should be introduced (as indicated with the brackets "«" and "»") and when vectors should be expressed in a new coordinate system (via a dot-product with a basis dyadic).

To illustrate the general appearance of the terms in Table 7, Table 8 shows some of the expressions formed for the satellite example. The many z intermediate variables shown are the result of the "« »"operation. The z terms are not defined in this paper (there are over 300 of them), since the intent here is just to give an idea of how expressions are formed for the various terms. However, the equations of motion for this system have been published elsewhere.[13]

(Some of the subexpressions shown in Table 8 have not been replaced with intermediate variables. Intermediate variables are not introduced by the computer algebra system unless they appear at least twice in the equations of motion. Thus, we know that expressions appearing in Table 8 did not occur more than once in the final equations and may not have appeared at all.)

Translational Terms

The approach used to develop an expression for central acceleration is similar to that used for angular acceleration and leads to the following:

$$a^{B*} = \frac{dv^{B*}}{dt} = \sum_{r=1}^{\nu}\left(\frac{du_r}{dt}v_r^{B*} + u_r\frac{dv_r^{B*}}{dt}\right) = \sum_{r=1}^{\nu} \dot{u}_r v_r^{B*} + a_{rem}^{B*} \tag{36}$$

Table 7 Formulas pertaining to rotational velocity and acceleration

Symbol	0 DOF	Rotor	1 DOF	3 DOF
$\omega^{A'}$		$\langle\langle\omega^A \cdot \vec{A}\rangle\rangle$	$\langle\langle\langle\omega^A \cdot \vec{A}\rangle\rangle \cdot \vec{B}\rangle\rangle$	
$^A\omega^B$		$u_r d_{rot}^{B}$[a]	$u_r d_{rot}^{B}$	
ω^B	ω^A	$\langle\langle\omega^{A'} + {}^A\omega^B\rangle\rangle$	$\langle\langle\omega^{A'} + {}^A\omega_t^B\rangle\rangle$	$u_{o+1}b_1 + u_{o+2}$ $b_2 + u_{o+3}b_3$[b]
$^A\omega_i^B$		$\begin{cases} d_{rot}^{B} \text{ for } i=r \\ 0 \text{ for } i \neq r \end{cases}$	$\begin{cases} d_{rot}^{B} \text{ for } i=r \\ 0 \text{ for } i \neq r \end{cases}$	
ω_i^B	ω_i^A	$\langle\langle\omega_i^A \cdot \vec{A}\rangle\rangle + {}^A\omega_i^B$	$\langle\langle\langle\omega_i^A \cdot \vec{A}\rangle\rangle \cdot \vec{B}\rangle\rangle$ $+ {}^A\omega_i^B$	$\begin{cases} b_{i-o} \text{ } i-o = 1,2,3 \\ 0 \text{ otherwise} \end{cases}$
α_{rem}^B	α_{rem}^A	$\langle\langle\alpha_{rem}^A \cdot \vec{A}$ $+ \omega^{A'} \times {}^A\omega^B\rangle\rangle$	$\langle\langle\alpha_{rem}^A \cdot \vec{B}$ $+ \omega^{A'} \times {}^A\omega^B\rangle\rangle$	0

[a] Speed u_r is the speed corresponding to the joint rotation of B.
[b] Index o is an offset to the indices of the three rotational speeds associated with B.

Table 8   Rotational velocity and acceleration terms for satellite example

Property	Body B	Body C	Body D	Body E	Body F
$\omega^{A\prime}$		$u_6b_3 + u_5b_2 + u_4b_1$	$Z_4c_1 + Z_6d_3 + Z_7d_2$	$u_6b_3 + u_5b_2 + u_4b_1$	$(u_4C_9 + u_5S_9)e_1 + Z_{15}f_3 + Z_{16}f_2$
$^A\omega^B$	0	u_7b_3	u_8c_1	u_9b_3	$u_{10}e_1$
ω^B	$u_4b_1 + u_5b_2$ $+ u_6b_3$	$u_5b_2 + u_4b_1$ $+ Z_3b_3$	$Z_6d_3 + Z_7d_2 + Z_{12}c_1$	$u_5b_2 + u_4b_1 + Z_{13}b_3$	$Z_{15}f_3 + Z_{16}f_2 + Z_{21}e_1$
$\omega_i^{A\prime}$		0, 0, 0, b_1, b_2, b_3, 0, 0, 0, 0	0, 0, 0, $(Z_8d_3 - Z_9d_2 + C_7c_1)$, $(-Z_{10}d_3 + Z_{11}d_2 + S_7c_1)$, $(C_8d_3 + S_8d_2)$, $(C_8d_3 + S_8d_2)$, 0, 0, 0	0, 0, 0, b_1, b, b_3, 0, 0, 0, 0	0, 0, 0, $(Z_{17}f_3 - Z_{18}f_2 + C_9e_1)$, $(-Z_{19}f_3 + Z_{20}f_2 + S_9e_1)$, $(C_{10}f_3 + S_{10}f_2)$, 0, 0, $(C_{10}f_3 + S_{10}f_2)$, 0
$^A\omega_i^B$	0, 0, 0, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, b_3, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, c_1, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, b_3, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, e_1
ω_i^B	0, 0, 0, b_1, b_2, b_3, 0, 0, 0, 0	0, 0, 0, b_1, b_2, b_3, b_3, 0, 0, 0	0, 0, 0, $(Z_8d_3 - Z_9d_2 + C_7c_1)$, $(-Z_{10}d_3 + Z_{11}d_2 + S_7c_1)$, $(C_8d_3 + S_8d_2)$, $(C_8d_3 + S_8d_2)$, c_1, 0, 0, $(C_8d_3 + S_8d_2)$, $(C_8d_3 + S_8d_2)$, c_1, 0, 0	0, 0, 0, b_1, b, b_3, 0, 0, b_3, 0	0, 0, 0, $(Z_{17}f_3 - Z_{18}f_2 + C_9$ $e_1)$, $(-Z_{19}f_3 + Z_{20}f_2 + S_9$ $e_1)$, $(C_{10}f_3 + S_{10}f_2)$, 0, 0, $(C_{10}f_3 + S_{10}f_2)$, e_1
α_{rem}^B	0	$-Z_{47}b_2 + Z_{48}b_1$	$Z_{51}c_1 + Z_{55}d_3 + Z_{56}d_2$	$-Z_{57}b_2 + Z_{58}b_1$	$Z_{59}e_1 + Z_{61}f_3 + (u_{10}Z_{15}$ $- Z_{60}C_{10})f_2$
α_{rem}^{AB}	0	$u_5u_7b_1 - u_4u_7b_2$	$u_8Z_6d_2 - u_8Z_7d_3$	$u_5u_9b_1 - u_4u_9b_2$	$u_{10}Z_{15}f_2 - u_{10}Z_{16}f_3$

Table 9   Formulas pertaining to translational velocity and acceleration[a]

Symbol	Nonrecursive	Recursive
r^{A*B*}		$\ll r^{A_0B_j} + \sum\limits_{i=1}^{N_{td}^B} q_{i+o}d_{ti}^B \gg$
$\omega^{B\prime}$	$\ll \omega^B \cdot \vec{B} \gg$	$\ll \omega^B \cdot \vec{B} \gg$
v_i^{A*B*}		$\begin{cases} d_{tj}^B \text{ for } i=o+j, j=1,\ldots,N_{td}^B \\ \ll\omega_i^A \cdot \vec{A}\gg \times r^{A*B_0} \quad \text{otherwise} \\ \quad + \ll\omega_i^B \cdot \vec{B}\gg \times r^{B_0B*} \end{cases}$
N_i^B	B	$\begin{cases} N_i^A \text{ if } v_i^{A*B*} = 0 \\ B \quad \text{otherwise} \end{cases}$
v_i^{B*}	$\begin{cases} b_{i-o}\, i \text{ is translational} \\ \quad \text{DOF of } B \\ 0 \quad \text{otherwise} \end{cases}$	$\ll(v_i^{A*} + v_i^{A*B*}) \cdot \vec{B}\gg$
$\vec{\vec{R}}^B$	$\ll(\alpha_{\text{rem}}^B \cdot \vec{B}) \times \vec{B}$ $+ \omega^{B\prime} \times (\omega^{B\prime} \times \vec{B})\gg$	$\ll(\alpha_{\text{rem}}^B \cdot \vec{B}) \times \vec{B}$ $+ \omega^{B\prime} \times (\omega^{B\prime} \times \vec{B})\gg$
$a_{\text{rem}}^{B\prime}$	$\ll\omega^{B\prime} \times \ll\sum\limits_{i=1}^{\nu} v_i^{B*}\gg - g \cdot \vec{B}\gg$	$\ll\left[a_{\text{rem}}^{A*} + \vec{\vec{R}}^A \cdot r^{A*B_0}\right.$ $+ \vec{\vec{R}}^B \cdot r^{B_0B*}$ $\left. + 2\sum\limits_{i=1}^{N_{td}^B} u_{o+j}(\ll\omega^A$ $\cdot \vec{A}\gg \times d_{tj}^B)\right] \cdot \vec{B}\gg$

[a] If B is "fixed" (see Table 3), substitute B_0 for $B*$. If A is fixed, substitute A_0 for $A*$.

where a_{rem}^{B*} is the central acceleration remainder, defined as

$$a_{\text{rem}}^{B*} = \sum_{i=1}^{\nu} u_i \frac{dv_i^{B*}}{dt} \tag{37}$$

As was the case for the rotational analysis, a component in v^{B*} that is an explicit function of time v_t^{B*} is required to be zero within the scope of this paper.

We again process the topology tree from the top down, such that the parent of each body B is analyzed before B is processed. To start, the ν partial angular velocities for N are defined as zero:

$$v_i^{N*} = 0 \qquad (i = 1, \ldots, \nu) \tag{38}$$

Also, the central acceleration remainder of N is set to the negative acceleration due to gravity:

$$a_{\text{rem}}^{N*} = -g \tag{39}$$

(If the system is not contained in a uniform gravitational field, as is the case for the satellite example, then the vector g is defined as zero.) Forces due to gravity are handled this way, rather than by defining a gravity force for each body, because it is much more efficient. All acceleration remainders that are derived using recursive relationships include gravity if it was included in N. For bodies down the tree, the effect of gravity is included without adding any complexity to the equations. Thus, when gravity is included, the effect on efficiency is usually that only a few multiplications are added to the equations of motion, regardless of the number of bodies.

Table 9 lists the formulas used to develop the corresponding terms for each body down the tree. As was the case for Table 7, the expressions at the bottom of the table refer to expressions defined at the top. Table 10 show some of these terms for the satellite example. [A few of the dyadic terms (in the row for $\vec{\vec{R}}^B$) are lengthy and are not shown in their entirety. All other terms are complete.]

Two expressions in the above table deserve discussion. For each body, a set of native bodies for the partial velocities is defined, whose elements are designated N_i^B, where $i = 1, \ldots, \nu$. Each of these identifies the body that is furthest up the tree in which the partial velocity is algebraically equal to the corresponding partial velocity of B. For example, in Table 10, N_1^D, N_2^D, and N_3^D are set to B, indicating that the first three partial velocities of D are algebraically equal to the corresponding partial velocities in B. On the other hand, the partial velocities for indices 4 through 8 are native to D. The significance of this information will be shown later.

Another expression that deserves comment is $\vec{\vec{R}}^B$, called an acceleration rotational dyadic. It accounts for the acceleration of a point due to angular rotation and angular acceleration of the body in which the point is fixed. The vector contribution to the translational acceleration of a point is obtained by projecting a position vector against this dyadic. It is used with the position vector going from the mass center of a parent body to the origin of the body of interest (r^{A*B_0}), and also with the position vector going from the origin to the mass center of the same body (r^{B_0B*}). As can be seen in the table, this dyadic can involve complex expressions. Computational savings are obtained by introducing intermediate variables for the scalars appearing in this dyadic.

Table 10 Translational velocity and acceleration terms for satellite example

Property	Body B	Body C	Body D	Body E	Body F
$r^{A^*B_0}$		$P_1 b_3$	$-(L_1 b_3 - L_3 c_2)$	$-(L_7 b_2 - P_1 b_3)$	0
$v_i^{B^*}$	$b_1, b_2, b_3, 0, 0, 0, 0, 0,$ $0, 0$	$(C_7 c_1 - S_7 c_2),$ $(C_7 c_2 + S_7 c_1), b_3,$ $-(Z_{22} c_1 + Z_{23} c_2),$ $(-Z_{22} c_2 + Z_{23} c_1), 0, 0,$ $0, 0, 0$	$(Z_8 d_3 - Z_9 d_2 + C_7 c_1)$ $(-Z_{10} d_3 + Z_{11} d_2$ $+ S_7 c_1), (C_8 d_3 + S_8 d_2),$ $(-Z_{25} d_3 + Z_{27} d_2$ $+ Z_{29} c_1), -(Z_{30} d_3$ $- Z_{31} d_2 + Z_{32} c_1),$ $- Z_{33} c_1, -Z_{33} c_1,$ $(L_6 d_2 - L_5 d_3), 0, 0$	$(C_9 e_1 - S_9 e_2),$ $(C_9 e_2 + S_9 e_1), b_3,$ $-(L_7 b_3 + Z_{34} e_1$ $+ Z_{35} e_2), (-Z_{34} e_2$ $+ Z_{35} e_1), (-Z_{36} e_2$ $+ L_7 C_9 e_1), 0, 0, 0, 0$	$(Z_{17} f_3 - Z_{18} f_2 + C_9 e_1),$ $(-Z_{19} f_3 + Z_{20} f_2$ $+ S_9 e_1), (C_{10} f_3 + S_{10} f_2),$ $-(Z_{37} f_2 + Z_{38} f_3 - Z_{39} e_1),$ $-(Z_{40} f_3 - Z_{41} f_2$ $+ Z_{42} e_1), (Z_{43} f_3 - Z_{44} f_2$ $+ Z_{46} e_1), 0, 0,$ $Z_{45} e_1, -L_8 f_3$
N_i^B	$B, B, B, B, B, B, B, B,$ B, B	$B, B, B, C, C, B, B, B,$ B, B	$B, B, B, D, D, D, D,$ D, B, B	$B, B, B, E, E, E, B, B,$ B, B	$B, B, B, F, F, F, B, B,$ F, F
$a_{\text{rem}}^{B^*}$	$Z_{65} b_1 + Z_{66} b_2 + Z_{67} b_3$	$-[Z_{74} c_2 - (Z_{73} C_7$ $+ Z_{71} S_7) c_1 +$ $- (Z_{67} - Z_{75}) b_3]$	$Z_{81} d_3 + Z_{82} d_2 + Z_{83} c_1$	$(Z_{84} C_9 + Z_{85} S_9) e_1$ $+ Z_{86} e_2 - Z_{87} b_3$	$Z_{90} f_2 - Z_{91} f_3 + Z_{92} e_1$
$\vec{R}^B$	$u_6 u_4[(b_3 b_1) + (b_1 b_3)]$ $+ u_4 u_5[(b_1 b_2) + (b_2 b_1)]$ $+ Z_{62}[(b_3 b_2) + (b_2 b_3)]$ $- (Z_{63} + u_6^2)(b_2 b_2)$ $- (Z_{63} + Z_{64})(b_3 b_3)$ $- (u_6^2 + Z_{64})(b_1 b_1)$	$-[-(-Z_{49} + Z_{50}$ $+ Z_{68}) (c_2 b_3) - Z_4$ $Z_5(c_1 c_2) - Z_4 Z_5(c_2 c_1)$ $- (Z_{49} - Z_{50} + Z_{68})$ $(b_3 c_2) + (Z_{69} + Z_5^2)(b_3 b_3)$ $+ \cdots]$	$(Z_7 Z_{12} - Z_{55})(c_1 d_2)$ $+ (Z_6 Z_{12} + Z_{56})(c_1 d_3)$ $+ (Z_{55} + Z_7 Z_{12})(d_2 c_1) +$ $- (Z_{56} - Z_6$ $Z_{12})(d_3 c_1) + - (Z_{51}$ $- Z_{76})(d_2 d_3) + \cdots$	$-[Z_{13}^2(e_1 e_1) + Z_{13}^2(e_2 e_2)$ $- Z_{13}(u_5 C_9 - u_4 S_9)$ $(e_2 b_3) - Z_{13}(u_4 C_9$ $+ u_5 S_9)(e_1 b_3) + (u_5 C_9$ $- u_4 S_9)^2(e_1 e_1) + \cdots]$	$-(Z_{61} - Z_{88})(e_1 f_2)$ $+ [(u_{10} Z_{15} - Z_{60} C_{10})$ $+ Z_{15} Z_{21}](e_1 f_3) + (Z_{61}$ $+ Z_{88})(f_2 e_1) + -[(u_{10} Z_{15}$ $- Z_{60} C_{10}) - Z_{15} Z_{21}]$ $(f_3 e_1) + \cdots$

Dynamical Equations

Once the terms in Tables 7 and 9 are obtained for all bodies, it is straightforward to finish the analysis to obtain the dynamical equations. By substituting Eqs. (32) and (36) into Eq. (12) and comparing with Eq. (2), the coefficient in the mass matrix for a particular row i and column j is obtained:

$$M_{ij} = \sum_B^{N_B} (\omega_j^B \cdot \overrightarrow{I}^{B^*} \cdot \omega_i^B + v_j^{B^*} \cdot v_i^{B^*} m^{B'}) \qquad (40)$$

where the symbol $m^{B'}$ is defined according to whether B is fixed or not:

$$m^{B'} = 0 \qquad B \text{ is fixed}$$

$$m^{B'} = m^{Bc} \qquad [\text{Eq. (15) otherwise}] \qquad (41)$$

Subtracting Eq. (40) from Eq. (12) yields the coefficient for element i in the force array (corresponding to row i in the mass matrix):

$$f_i = \sum_B^{N_B} \left[\left(\sum_{t=1}^{N_{B,T}} T_t^B - \alpha_{\text{rem}}^B \cdot \overrightarrow{I}^{B^*} - \omega^B \times \overrightarrow{I}^{B^*} \cdot \omega^B \right) \cdot \omega_i^B + \left(\sum_{f=1}^{N_{B,F}} F_f^B - a_{\text{rem}}^{B^*} m^{B'} \right) \cdot v_i^{B^*} \right] \qquad (42)$$

To obtain efficient equations without extensive symbolic manipulation, it is recommended that intermediate variables be introduced as follows. For the mass-matrix coefficient, two cases exist. The first applies when either the native body N_i^B or N_j^B is set to B. In such a case, the following formulation is suggested:

$$M_{ij} = \sum_B^{N_B} (\omega_j^B \cdot « \overrightarrow{I}^{B^*} \cdot \omega_i^B » + v_j^{B^*} \cdot « v_i^{B^*} m^{B'} »)$$

$$(N_i^B = B \text{ or } N_j^B = B) \qquad (43)$$

Otherwise, the recommended formulation is

$$M_{ij} = \sum_B^{N_B} (\omega_j^B \cdot « \overrightarrow{I}^{B^*} \cdot \omega_i^B » + «\hat{v}_j^{B^*} \cdot \hat{v}_i^{B^*} » m^{B'})$$

$$(N_i^B \neq B \text{ and } N_j^B \neq B) \qquad (44)$$

In the case of Eq. (44), the caret symbol ($\char`^$) appears over the partial velocities, indicating that they are taken from the body in which they were introduced rather than from B. The difference is that the partial velocities for B are in the basis of B, as needed for the recursive relations of Table 9. To see the significance of this information, compare the first partial velocity for all five bodies of the example in Table 10. All five sets are expressed differently in the table, yet all are algebraically equivalent, being simply b_1, b_2, and b_3. That is,

$$v_1^{B^*} \cdot v_2^{B^*} = v_1^{C^*} \cdot v_2^{C^*} = v_1^{D^*} \cdot v_2^{D^*} = v_1^{E^*} \cdot v_2^{E^*} = v_1^{F^*} \cdot v_2^{F^*} = 0 \quad (45)$$

Equation (45) is true numerically for any valid set of generalized coordinates and speeds. It is also true symbolically for the expressions listed in the table for bodies B, C, and E. However, for bodies D and F, the necessary symbolic cancellations will not occur because intermediate z variables have been substituted for the complicated trigonometric expressions. The formulation for the force array is as follows:

$$f_i = \sum_B^{\text{all bodies}} \left[« \left(\sum_{t=1}^{N_{B,T}} T_t^B \right) » \cdot \omega_i^B - «(\alpha_{\text{rem}}^B \cdot \overrightarrow{I}^{B^*} + \omega^B \times \overrightarrow{I}^{B^*} \cdot \omega^B)» \cdot \omega_i^B + « \left(\sum_{f=1}^{N_{B,F}} F_f^B \right) » \cdot \hat{v}_i^{B^*} - a_{\text{rem}}^{B^*} \cdot «v_i^{B^*} m^{B'}» \right] \qquad (46)$$

Note that the partial velocity dotted with the applied forces is expressed in its original basis (as indicated with the caret), which is either B or a body up the tree from B.

The set of dynamical equations presented as Eq. (2) can be uncoupled symbolically using lower-upper (LU) triangular decomposition. By introducing intermediate variables, the solution can be guaranteed to involve no more computation than a numerical solution. However, if there is any sparsity in the mass matrix, the symbolic solution is more efficient than the numerical one. The locations of the zeros in the mass matrix

are shown below 1) when the equations are ordered by index $(1, 2, \ldots, 10)$, and 2) with the equations ordered to move the zeros to the upper-left corner of the matrix (the permutation is done to avoid matrix fill when the matrix is decomposed):

$$M = \begin{bmatrix} \cdot & 0 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot & \cdot & \cdot & 0 & \cdot & 0 & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 \\ \cdot & \cdot & 0 & \cdot & \cdot & \cdot & 0 & 0 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 & 0 & \cdot \end{bmatrix} \rightarrow \begin{bmatrix} \cdot & 0 & 0 & \cdot & 0 & 0 & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & 0 & 0 & \cdot & \cdot & 0 & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & 0 & 0 & \cdot & \cdot & \cdot & 0 & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & 0 & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & 0 & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & 0 & \cdot & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \qquad (47)$$

Discussion

The formalism presented earlier was developed with two objectives in mind: 1) to support the development of software that automates the formulation and programming of the complete equations of motion for multibody systems, and 2) to obtain highly efficient computer code for solving those equations. The second objective requires that symbolic computation be used to take advantage of special geometric simplifications that are unique to a particular system. (For example, all of the rotation axes in the spacecraft are orthogonal.) While it is true that symbolic multibody programs have been developed earlier and are in use, they deal mainly with inertial forces and torques. None can automatically incorporate active forces and moments into the equations: it is up to the analyst to provide these terms independently. For many everyday dynamical systems, such as ground vehicles, the terms due to active forces and moments are much more complicated than the inertial terms. The basis-free vector representation used in the formalism presented here permits any imaginable force or moment to be included correctly and automatically by software based on the formalism.

The three parts of the analysis (system description, kinematics, and dynamics) are independent of each other. Thus, it is possible to extend or refine analysis capabilities in one area of the analysis without reworking the others. For example, the possible modeling simplifications made when bodies are added are unrelated to the method used to obtain kinematical and dynamical equations. Similarly, the kinematics analysis method is valid regardless of how the dynamical equations are formed. Although this paper is limited in scope to holonomic tree-topology systems, the basic formalism has been extended to deal with nonholonomic systems by changing only the dynamics part of the analysis.[13] A further extension, to include closed kinematical loops, was also made and required only a change to the kinematical part of the analysis.[13] (These extensions will be detailed in a future paper.) The methods described in this paper have been programmed in Lisp and are part of a software package called AUTOSIM, developed at the University of Michigan to automatically generate simulation codes for multibody systems. The inputs provided by a dynamicist to describe the spacecraft example in AUTOSIM are listed in the companion paper.[12]

Many of the details of the formalism are rules of thumb that would be performed by an experienced dynamicist deriving equations by hand. By paying attention to many minor details, including forced coordinate transformations and the introduction of intermediate variables, highly efficient computational code is obtained. As a point of reference, Table 11 summarizes the efficiency of the code obtained by AUTOSIM and compares it with results from another symbolic multibody program SD/FAST.[9]

When forming equations with pencil and paper, dynamicists like to throw out terms that are known to be numerically negligible. Also, truncated Taylor expansions are substituted for trigonometric functions of angles that are known to be small (that is, $\sin x \approx x$; $\cos x \approx 1$). For example, if the satellite model is intended to apply for situations in which its attitude

Table 11 Computational efficiency of spacecraft code

Source	Adds and subtracts	Multiplies, divides, and function cells
SD/FAST user's manual	709	1094
AUTOSIM, using full, nonlinear formulation	628	791
AUTOSIM, using small variables for 8 DOF	442	514

deviates only slightly from the nominal orientation, the three rotations of the body (q_4, q_5, and q_6) can be modeled as small. Also, the two rotations of the flexible boom (q_9 and q_{10}) are always small. Table 11 shows the effect of these assumptions on the operation counts. The alternate formulation is made by applying the same analysis method but letting the computer algebra system simplify expressions involving small quantities. For an example slew maneuver, identical numerical results were obtained with the two formulations.[12]

The formalism from this paper includes many options that are intended to simplify equations of motion. The significance of these options depends greatly on the topology of the multibody system being considered. For example, in a robot system in which all bodies are connected by hinges, there are no simplifications to be made by the use of native partial velocities. However, when applied to vehicle systems, the attention to native bodies can yield significant simplifications. On the other hand, the highly recursive relationships and the rotation dyadic in Tables 7 and 9 yield compact robot equations that are efficient compared to other formulations but offer little improvement for vehicle models.

Conclusions

A multibody formalism has been presented that includes the sort of judgments a human analyst makes in formulating equations of motion for a tree-topology multibody system. To use the formalism, a dynamicist describes the multibody system in geometric terms, using vectors to specify allowable motions of each body relative to another body in the system. Forces and torques acting on each body are specified as vectors to permit the inclusion of any conceivable force- or torque-producing behavior. From the description of the multibody system, the formalism has rules to determine how generalized coordinates and speeds are defined. (The speeds are not necessarily the derivatives of the coordinates.) The formalism includes a number of features that distinguish it from previously reported work: 1) it does not keep track of coordinate systems that were used to define vectors, 2) when appropriate, it forces vectors of an unknown nature into predetermined vector bases by dotting them with dyadics, 3) it specifies when "intermediate variables" are to be introduced to improve computational efficiency, 4) it defines speeds using both recursive and nonrecursive definitions, depending on topology, 5) it includes a method for deriving kinematical equations when the speeds are not the derivatives of the generalized coordinates, 6) it uses the parallel axis theorem to lump inertial properties

of bodies together when possible, 7) it uses a method derived from Kane's work to develop the dynamical equations, modified to take full advantage of recursion through the use of a rotation dyadic, and 8) it symbolically uncouples the implicit equations to exploit sparsity in the mass matrix. There are significant benefits deriving from these features. The formalism permits the dynamicist a great deal of flexibility in describing a model, and at the same time, the input description to the formalism is very simple. Yet the computer codes generated are highly efficient. Another benefit, reported elsewhere, is that the relative independence of the analysis stages allow easy extension to multibody systems that have nonholonomic constraints and/or closed kinematical loops.

Acknowledgments

References
[1]Orlandea, N., Chace, M. A., and Calahan, D. A., "A Sparsity-Oriented Approach to the Dynamic Analysis and Design of Mechanical Systems, Parts I and II," *Journal of Engineering for Industry,* Vol. 99, Aug. 1977, pp. 773-784.

[2]Nikravesh, P. E., and Haug, E. J., "Generalized Coordinate Partitioning for Analysis of Mechanical Systems with Nonholonomic Constraints," *ASME Journal of Mechanisms, Transmissions, and Automation in Design,* Vol. 105, Sept. 1983, pp. 379-384.

[3]Frisch, H. P., "A Vector-Dyadic Development of the Equations of Motion for *N*-Coupled Rigid Bodies and Point Masses," Goddard Space Flight Center, D-7767, Greenbelt, MD, Oct. 1974.

[4]Wittenburg, J., *Dynamics of Systems of Rigid Bodies,* B.G. Teubner, Stuttgart, Germany, 1977.

[5]Schiehlen, W. O., "Dynamics of Complex Multibody Systems," *Solid Mechanics Archives,* Vol. 9, No. 2, 1984, pp. 159-195.

[6]Roberson, R. E., and Schwertassek, R., *Dynamics of Multibody Systems,* Springer-Verlag, Berlin, Germany, 1988.

[7]Kane, T. R., and Levinson, D. A., "Formulation of Equations of Motion for Complex Spacecraft," *Journal of Guidance and Control,* Vol. 3, No. 2, 1980, pp. 99-112.

[8]Kane, T. R., and Levinson, D. A., *Dynamics, Theory and Applications,* McGraw-Hill Series in Mechanical Engineering, McGraw-Hill, New York, 1985.

[9]Rosenthal, D. E., and Sherman, M. A., "High Performance Multibody Simulations via Symbolic Equation Manipulation and Kane's Method," *Journal of the Astronautical Sciences,* Vol. 34, No. 3, 1986, pp. 223-239.

[10]Rosenthal, D. E., "Triangularization of Equations of Motion for Robotic Systems," *Journal of Guidance, Control, and Dynamics,* Vol. 11, No. 3, 1988, pp. 278-281.

[11]Featherstone, R., *Robot Dynamics Algorithms,* Kluwer International Series in Engineering and Computer Science Robotics: Vision, Manipulation, and Sensors, Kluwer, Boston, MA, 1987.

[12]Sayers, M. W., "Symbolic Computer Language for Multibody Systems," *Journal of Guidance, Control, and Dynamics,* Vol. 14, No. 6, 1991, pp. 1153-1163.

[13]Sayers, M. W., "Symbolic Computer Methods to Automatically Formulate Vehicle Simulation Codes," Ph.D. Dissertation, University of Michigan, Ann Arbor, MI, 1990.