JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 20, No. 4, July–August 1997

Order *N* Formulation for Flexible Multibody Systems in Tree Topology: Lagrangian Approach

S. Pradhan* and V. J. Modi†
University of British Columbia, Vancouver, British Columbia V6T 1Z4, Canada
and
A. K. Misra‡
McGill University, Montreal, Quebec H3A 2K6, Canada

An order *N* algorithm is developed for dynamic simulation of flexible multibody systems in tree topology with open as well as closed branches. The joints between the bodies permit slew and translation degrees of freedom. Application of the chain rule of differentiation to the Lagrange equations of motion in conjunction with position and velocity transformations leads to the factorized mass matrix and an order *N* algorithm for the forward dynamics computation. The constrained degrees of freedom are incorporated by the Lagrange multipliers. Computation of the forward dynamics of flexible multibody systems using a Lagrangian order *N* method represents an innovation of considerable importance. An example of the space-platform-based mobile manipulator system complements the development.

I. Introduction

THE development of a relatively general formulation is receiving increased attention as the dynamic systems become larger and more complex. The computational cost involved in the numerical simulation of such systems also becomes an important factor for the present day systems. Mathematical representation of such flexible multibody systems leads to a set of equations of the form

$$[M(q,t)]\{\ddot{\boldsymbol{q}}\} + \{F(q,\dot{q},t)\} = 0$$

where $[M(q,t)]$ is the system mass matrix, $\{F(q,\dot{q},t)\}$ the forcing function, and $\{\boldsymbol{q}\}$ the vector of generalized coordinates. The forward dynamics computation using the equation $\{\ddot{\boldsymbol{q}}\} = -[M(q,t)]^{-1}\{F(q,\dot{q},t)\}$ requires $\mathcal{O}(N^3)$ arithmetic operations, where N is the number of bodies constituting the system. Thus, the computational cost for the dynamics simulation of a multibody system grows rapidly with N.

Reduction of the computation time for simulation of systems with a large number of bodies is an important issue in the field of multibody dynamics. To that end, several algorithms have been proposed over the years.[1–8] In these approaches, the computational cost involved in formulating and inverting the mass matrix is proportional to N and, hence, the designation $\mathcal{O}(N)$ algorithm. Obviously, these methods require far fewer computational steps than an $\mathcal{O}(N^3)$ formulation, particularly when N is large. The formulations, except the one by Menon and Kurdila,[7] use recursive approaches based on the fact that the acceleration of a body in a chain can be expressed in terms of accelerations of the preceding bodies. Similarly, the constraint forces at a joint can also be written in terms of those at the succeeding joints.

A variety of well-known approaches, available for formulation of the problem,[9] have been used to develop $\mathcal{O}(N)$ algorithms. For example, Hollerbach[1] uses the Lagrangian equations, Bae and Haug[2] employ the method of virtual work, Roberson and Schwertassek[3] as well as van Woerkom and de Boer[4] work with the Newton–Euler method, Keat[5] introduces a velocity transform method, Rosenthal[6] uses Kane's equation, Menon and Kurdila[7] approach the problem through the range space method, whereas Rodriguez and Kreutz-Delgado[8] use the filtering and smoothing approach of optimal estimation.

A number of computer codes are commercially available to simulate flexible multibody systems. They include ALLFLEX, Treetops, and others. In ALLFLEX, developed by Ho and Herber,[10] both the Lagrangian and Newtonian approaches are used to formulate the equations of motion. The direct path method is used to develop the complete kinematic relations. No special consideration is given for improving the computational efficiency of the formulation, and it is an $\mathcal{O}(N^3)$ algorithm. The program developed by Singh et al.,[11] called Treetops, develops the equations of motion using Kane's method. This is also an $\mathcal{O}(N^3)$ algorithm. It can simulate systems in tree topology defined as an arbitrary set of rigid and flexible bodies connected by hinges.

The present paper proposes a different approach to the $\mathcal{O}(N)$ algorithm for the dynamic simulation of flexible multibody systems in tree topology including closed loops. The formulation can model most of the current and future space-based systems, e.g., space station with a dexterous mobile manipulator, tethered satellite system, etc., as well as ground-based manipulators. Moreover, there is no formulation reported in the open literature that studies the forward dynamics of multibody systems using a Lagrangian order N algorithm. The only effort in that direction is by Hollerbach,[1] who studies the inverse dynamics of a special case involving rigid system in the chain topology.

The inverse dynamics problem has been studied extensively in robotics. It involves computation of the mass matrix and forcing function terms required for controller implementation using computed torque technique. However, for dynamic simulation (also called forward dynamics study), it is required to solve for the acceleration vector. In many approaches, this involves the computation of mass matrix and forcing functions, as well as the inversion of the mass matrix. Hollerbach[1] developed a Lagrangian-approach-based $\mathcal{O}(N)$ algorithm to compute the mass matrix and forcing functions, i.e., inverse dynamics, for a system having rigid bodies in chain topology, whereas the current approach solves for the acceleration vector, i.e., forward dynamics, for flexible systems in tree configuration. Thus, the problems addressed and approaches used are rather different. Furthermore, extension of Hollerbach's algorithm to solve a forward dynamics problem of a flexible multibody system in a tree topology would be a formidable task.

In the present formulation, application of chain rule of differentiation to the Lagrange equation in conjunction with appropriate velocity and position transformations leads to a suitable factorization

Received June 24, 1996; revision received Feb. 12, 1997; accepted for publication March 19, 1997.

*Postdoctoral Fellow, Department of Mechanical Engineering; currently Control Systems Engineer, Bristol Aerospace Ltd., Winnipeg, Manitoba R3C 2S4, Canada. Member AIAA.

†Professor Emeritus, Department of Mechanical Engineering. Fellow AIAA.

‡Professor, Department of Mechanical Engineering. Associate Fellow AIAA.

of the mass matrix and the subsequent development of the $\mathcal{O}(N)$ algorithm. The constrained degrees of freedom are incorporated by Lagrange multipliers, which are solved recursively. The essentially nonrecursive nature of the present algorithm is an advantage and makes this approach attractive for implementation through parallel computation. The Lagrangian approach is also attractive as the internal constraint forces are accounted for implicitly. On the other hand, explicit expressions for the kinetic and potential energies are obtained, which can be used to verify the formulation. The simplicity in derivation and implementation, as well as its versatility, makes the algorithm a useful tool in studying a large class of flexible multibody system.

The paper is arranged as follows. To begin, several modeling issues are discussed. This is followed by the development of the proposed $\mathcal{O}(N)$ algorithm. The example of the space-station-based mobile remote manipulator system (MRMS) is considered to provide better appreciation of the algorithm. The paper ends with some concluding comments based on the analysis.

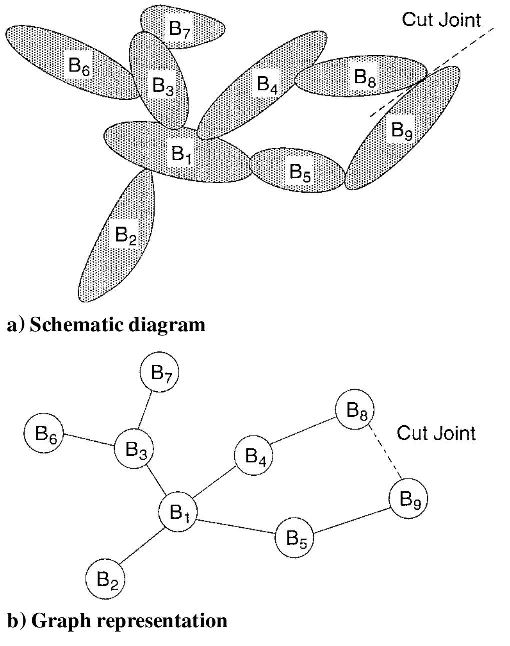
II. Kinematics and Other Modeling Issues

A. System Description

Consider a generic system of N flexible bodies connected in an arbitrary configuration (Fig. 1a). The joints between the bodies can have up to six degrees of freedom. Let G be the graph representing the system (Fig. 1b), where bodies are nodes (vertices) and joints are links (edges).[12] Let the graph be rooted at node one, i.e., body one. If G has closed loops, introduce a minimum number of joint cuts to eliminate them. The graph with cut joints is denoted by G'.

Let the symbols B, J, and J^c denote bodies, uncut joints, and cut joints, respectively. The graph G', generated by deleting J^c from G, has either open chain or open tree topology. The bodies, i.e., nodes, are numbered as B_j, $j = 1, 2, \dots, N$, in a scheme shown in Fig. 1. Consider B_1 to be the root of the tree. If, when passing from B_1 to an arbitrary body B_j on G', one goes through B_k, then B_k is said to be inboard of B_j and B_j is said to be outboard of B_k. Let J_j be the uncut joint connecting B_j to its immediate inboard body. This joint carries the actuators to move B_j with respect to its inboard body (B_{i_j}).

In G', the bodies are numbered in such a way that the number of the inboard body of B_j is always less than j. In addition, the number of any peripheral body in G', i.e., body without any outboard body, is always greater than that of any internal body. This numbering scheme ensures that the transformation matrices used in latter sections are lower block triangular. Let S_j^i be the integer set containing the number of all bodies inboard of B_j and $S_j = (S_j^i, j)$.



a) Schematic diagram

b) Graph representation

Fig. 1 Multibody system in tree topology.

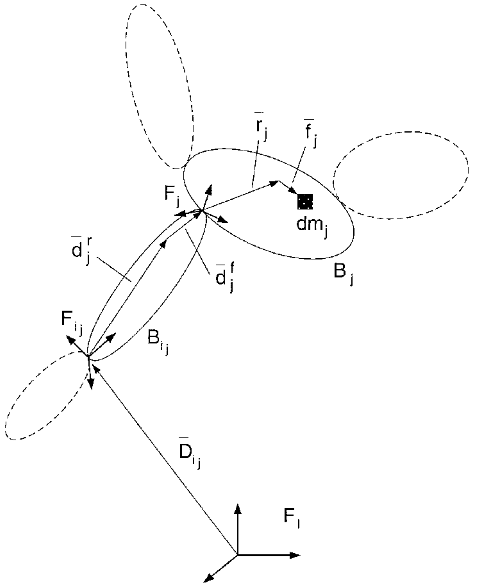


Fig. 2 Coordinate frames and vectors used to define an elemental mass.

The number of the body immediately inboard of B_j is denoted by i_j. Define I to be the set of N integers whose first element is zero and jth element ($j > 1$) is i_j. Let o_{kj} denote the number of the body immediately outboard of B_k in a path from B_1 to B_j.

The system is free to follow an arbitrary trajectory. B_1 is free to undergo three-dimensional rotations. The motion of B_j, $j = 2, \dots, N$, can be represented by translation and rotation with respect to B_{i_j}. The joint J_j, $j = 2, \dots, N$, can be modeled as springs and dampers for both translating and slewing degrees of freedom.

B. Coordinate Frames

Several coordinate frames are selected to express the motion of any elemental mass in the system (Fig. 2). The motion of the entire system is expressed with respect to an inertial frame F_I. The motion of B_j, $j = 2, \dots, N$, is expressed with respect to the frame F_j located at J_j. The motion of B_1 is expressed with respect to a body fixed coordinate system F_1 located at an arbitrary point on the first body. The axes x_j, y_j, and z_j of the frame F_j can be oriented along any arbitrary orthogonal directions. The relative orientation of each frame F_j with respect to the inertial frame F_I is specified by a set of three Euler angles α_j, β_j, and γ_j.

Let $\boldsymbol{D}_j$ be the position vector of the origin of the frame F_j with respect to F_I, $\boldsymbol{d}_j$ ($\boldsymbol{d}_j^r + \boldsymbol{d}_j^f$) the position vector of the origin of F_j with respect to F_{i_j} (Fig. 2), $\boldsymbol{d}_j^r$ the rigid component of $\boldsymbol{d}_j$, and $\boldsymbol{d}_j^f$ the flexible component of $\boldsymbol{d}_j$. With these notations, the position vector $\boldsymbol{D}_j$ can be written as

$$\boldsymbol{D}_j = \sum_{k \in S_j} \boldsymbol{d}_k = \sum_{k \in S_j} \left(\boldsymbol{d}_k^r + \boldsymbol{d}_k^f\right), \qquad j = 1, \dots, N \tag{1}$$

with $\boldsymbol{d}_1^f = 0$. This equation can be expressed recursively:

$$\boldsymbol{D}_1 = \boldsymbol{d}_1^r \tag{2a}$$

$$\boldsymbol{D}_j = \boldsymbol{D}_{i_j} + \boldsymbol{d}_j^r + \boldsymbol{d}_j^f, \qquad j = 2, \dots, N \tag{2b}$$

For convenience, the vectors and matrices in rest of the paper are presented as { } and [ ], respectively.

C. Structural Modeling

Flexibility of present-day engineering structures, particularly space-based systems, is a natural consequence of a need for large structures that should have a high strength-to-weight ratio. There are several approaches to describe the flexible character of a structure, including the use of the finite element method,[13] Rayleigh–Ritz approach,[14] substructuring or component mode synthesis,[15] and others. The most commonly used procedure is the method of assumed modes. Here, a finite number of component level shape functions, which are taken to be admissible functions and satisfy only the geometric boundary conditions, are used to develop the ordinary differential equations for the structural vibrations.[14,16]

Modeling of geometric stiffening effects is an important issue in the dynamic study of flexible bodies undergoing large motions. Since the first publication on this subject by Kane et al.,[17] a number of modeling approaches have been proposed, which are reviewed in detail by Sharf.[18] In the study by Kane et al.,[17] the stiffening effect is incorporated through the use of stretch instead of axial displacement as the generalized coordinate. The stretch is expressed as a nonlinear function of the orthogonal elastic deformations. In a comment to the study by Kane et al., Hanagud and Sarkar[19] have shown that the stiffening effects can be modeled by the conventional approach where the deformation field can be described in terms of an orthogonal set of elastic displacements (f_x, f_y, and f_z). Here, the nonlinear strain-displacement relation has to be used to include the stiffening effect. Based on the comparison of different approaches, Sharf[18] concludes that the conventional approach combined with the displacement formulation, i.e., nonlinear strain-displacement relation, of the strain energy is the most accurate and general. One does not require an approximation of foreshortening or the stress state of the body, the two major advantages in its application to the study of multibody dynamics. This approach is used in the present study to model the flexible elements of the system.

The components of most of the multibody systems can be represented by string, beam, membrane, plate, and shell type structural elements. The flexible deformation vector $\{\boldsymbol{f}_j\}$ of any elemental mass in the jth body can be expressed as a vector consisting of deflections in three orthogonal directions:

$$\{\boldsymbol{f}_j\} = \left\{\begin{matrix} f_{x_j} \\ f_{y_j} \\ f_{z_j} \end{matrix}\right\} \tag{3}$$

where f_{x_j}, f_{y_j}, and f_{z_j} are flexible deformations along the x_j, y_j, and z_j directions, respectively. Using the assumed mode method, the deformations of the jth body at an arbitrary point can be expressed as

$$f_{k_j} = \sum_{l=1}^{n_{k_j}} \Phi^l_{k_j}(r_j)\delta^l_{k_j}(t), \qquad k = x, y, z \tag{4}$$

where $\{\boldsymbol{r}_j\}$ is the position vector of the point with respect to F_j, $\Phi^l_{k_j}(r_j)$ the lth mode function, $\delta^l_{k_j}(t)$ the lth flexible generalized coordinate, and n_{k_j} the number of modes used to discretize the elastic deformation along the direction k_j ($k = x, y, z$). Now Eq. (3) can be expressed as

$$\{\boldsymbol{f}_j\} = [\Phi_j]\{\delta_j\} \tag{5}$$

where

$$[\Phi_j] = \begin{bmatrix} \left\{\Phi_{x_j}(r_j)\right\}^T & O & O \\ O & \left\{\Phi_{y_j}(r_j)\right\}^T & O \\ O & O & \left\{\Phi_{z_j}(r_j)\right\}^T \end{bmatrix} \in \Re^{3\times n_j}$$

$$\{\delta_j\} = \left\{\begin{matrix} \left\{\delta_{x_j}\right\} \\ \left\{\delta_{y_j}\right\} \\ \left\{\delta_{z_j}\right\} \end{matrix}\right\} \in \Re^{n_j}, \qquad \left\{\Phi_{k_j}(r_j)\right\} = \left\{\begin{matrix} \Phi^1_{k_j}(r_j) \\ \Phi^2_{k_j}(r_j) \\ \vdots \\ \Phi^{n_{k_j}}_{k_j}(r_j) \end{matrix}\right\} \in \Re^{n_{k_j}}$$

$$\left\{\delta_{k_j}\right\} = \left\{\begin{matrix} \delta^1_{k_j}(t) \\ \delta^2_{k_j}(t) \\ \vdots \\ \delta^{n_{k_j}}_{k_j}(t) \end{matrix}\right\} \in \Re^{n_{k_j}}, \qquad n_j = \sum_k n_{k_j}, \qquad k = x, y, z$$

and O is the null matrix of appropriate dimension. The mode shape functions depend on the element (beam, plate, or string) under consideration and the boundary conditions used. The effect of foreshortening of beams can also be accounted for in the flexibility modeling.

Structural damping is an inevitable source of energy dissipation during vibration. With a constant damping ratio for all modes, high-frequency vibrational modes damp out quickly. As a result, the inherent infinite (or very large) dimensional problems become practically finite, and perhaps even of modest order, thus making the control easier. Therefore, the modeling of structural damping is quite crucial in the dynamic formulation. Structural damping is often approximated as directly proportional to frequency for harmonic analysis. It is also represented as a complex elastic modulus with the imaginary part a small fraction, e.g., 0.01, of the strain modulus.[20] The energy dissipation due to structural damping can be approximated by Rayleigh's dissipation function, and then the damping effects can be modeled by the Lagrange equation.[21] There are several other issues related to the structural modeling, e.g., flexibility effects on design, control algorithm, sensor–actuator placement, etc. They are not within the scope of the present paper and are discussed by other researchers.[22]

D. Velocity Transformation

In the system under consideration (Fig. 1), each body can translate with respect to its neighboring bodies, go through rotations in three dimensions, and vibrate along three orthogonal directions. To represent these motions, the generalized coordinates for the jth body consist of the vector $\{\boldsymbol{d}^r_j\}$; the Euler angles α_j, β_j, and γ_j; and the flexible coordinates [Eq. (5)], $\{\delta_j\}$. Let $\{\boldsymbol{q}_j\}$ be the vector of generalized coordinates for B_j. Then,

$$\{\boldsymbol{q}_j\} = \left\{\begin{matrix} \left\{d^r_j\right\} \\ \{\eta_j\} \\ \{\delta_j\} \end{matrix}\right\} \tag{6}$$

where $\{\eta_j\} = \{\alpha_j, \beta_j, \gamma_j\}^T$.

The development of equations of motion by directly using $\{\boldsymbol{q}_j\}$, $j = 1, \dots, N$, requires order N^3 number of operations. To develop an efficient order N algorithm, a coordinate transformation is used. The corresponding velocity transformation is similar to that proposed by Jerkovsky[23] and also used by Keat.[5] However, in the present study, the resulting kinematic relations and the order N algorithm are quite different. The formulation by Keat[5] uses only one form of velocity transformation and develops a recursive algorithm. However, the present approach uses two different forms of the velocity transformation, which results in a nonrecursive algorithm. Furthermore, the present order N formulation is based on the Lagrangian procedure, which has not been addressed so far by other researchers.

Let the transformed coordinate vector of B_j be denoted by $\{\boldsymbol{q}^t_j\}$, defined as

$$\left\{\boldsymbol{q}^t_j\right\} = \left\{\begin{matrix} \{\boldsymbol{D}_j\} \\ \{\eta_j\} \\ \{\delta_j\} \end{matrix}\right\} \tag{7}$$

where $\{\boldsymbol{D}_j\}$ is the position vector to the origin of the frame F_j with respect to the inertial frame [Eq. (1)]. Then, the relation between the original and transformed generalized velocities can be expressed as

$$\left\{\dot{\boldsymbol{q}}^t_1\right\} = [R_1]\{\dot{\boldsymbol{q}}_1\} \tag{8a}$$

$$\left\{\dot{\boldsymbol{q}}^t_j\right\} = \left[R^n_{ji_j}\right]\left\{\dot{\boldsymbol{q}}^t_{i_j}\right\} + [R_j]\{\dot{\boldsymbol{q}}_j\}, \qquad j = 2, \dots, N \tag{8b}$$

where

$$\left[R^n_{ji_j}\right] = \begin{bmatrix} [U] & \left[P_{i_j}(d_j)\right] & \left[T_{i_j}\right]\left[\Phi_{i_j}\left(\boldsymbol{d}^r_j\right)\right] \\ O & O & O \end{bmatrix} \in \Re^{N_j \times N_{i_j}}$$

$$[R_1] = [U] \in \Re^{N_1 \times N_1}$$

$$[R_j] = \begin{bmatrix} \left[T_{i_j}\right] + \left[Q_{i_j}\left(\boldsymbol{d}^r_j\right)\right] & O \\ O & [U] \end{bmatrix} \in \Re^{N_j \times N_j}, \qquad j = 2, \dots, N$$

$$[P_k(\boldsymbol{d}_j)] = \left[\left[T_{\alpha_k}\right]\{\boldsymbol{d}_j\} \;\vdots\; \left[T_{\beta_k}\right]\{\boldsymbol{d}_j\} \;\vdots\; \left[T_{\gamma_k}\right]\{\boldsymbol{d}_j\}\right] \in \Re^{3\times 3}$$

$$\left[T_{l_k}\right] = \left[\frac{\partial T_k}{\partial l_k}\right], \qquad l = \alpha, \beta, \gamma$$

$$\left[Q_{i_j}\left(\boldsymbol{d}^r_j\right)\right] = \left[\left\{Q^x_{i_j}\right\} \;\vdots\; \left\{Q^y_{i_j}\right\} \;\vdots\; \left\{Q^z_{i_j}\right\}\right] \in \Re^{3\times 3}$$

$$\left\{Q^k_{i_j}\right\} = \left[T_{i_j}\right]\left[\frac{\partial\left[\Phi_{i_j}\left(d^r_j\right)\right]}{\partial d^r_{k_j}}\right]\left\{\delta_{i_j}\right\}, \qquad k = x, y, z$$

$$\{\boldsymbol{d}_j\} = \left\{\boldsymbol{d}^r_j\right\} + \left\{\boldsymbol{d}^f_j\right\} = \begin{Bmatrix} \boldsymbol{d}^r_{x_j} \\ \boldsymbol{d}^r_{y_j} \\ \boldsymbol{d}^r_{z_j} \end{Bmatrix} + \left[\Phi_{i_j}\left(\boldsymbol{d}^r_j\right)\right]\left\{\delta_{i_j}\right\} \in \Re^3$$

where

$[U]$ = unit matrix of appropriate dimension
O = null matrix of appropriate dimension
$[T_j]$ = transformation matrix between frames F_j and F_I
N_j = total number of generalized coordinates of B_j, $n_j + 6$

Stacking and rearranging Eq. (8) for all B_j yields

$$\{\dot{\boldsymbol{q}}^t\} = [U - R^n]^{-1}[R]\{\dot{\boldsymbol{q}}\} \tag{9}$$

where

$$\{\dot{\boldsymbol{q}}^t\} = \begin{Bmatrix} \dot{\boldsymbol{q}}^t_1 \\ \dot{\boldsymbol{q}}^t_2 \\ \vdots \\ \dot{\boldsymbol{q}}^t_N \end{Bmatrix} \in \Re^{N_g}, \qquad \{\dot{\boldsymbol{q}}\} = \begin{Bmatrix} \dot{\boldsymbol{q}}_1 \\ \dot{\boldsymbol{q}}_2 \\ \vdots \\ \dot{\boldsymbol{q}}_N \end{Bmatrix} \in \Re^{N_g}$$

$$[R] = \begin{bmatrix} [R_1] & O & \cdots & O \\ O & [R_2] & \cdots & O \\ \vdots & \vdots & \ddots & \vdots \\ O & O & \cdots & [R_N] \end{bmatrix} \in \Re^{N_g \times N_g}$$

and $[R^n] \in \Re^{N_g \times N_g}$, for $j = 1, \ldots, N$ and $k = 1, \ldots, N$, is defined as

$$(j, k)\text{th block of } [R^n] = \begin{cases} O, & \text{if} \quad k \neq i_j \\ \left[R^n_{ji_j}\right], & \text{if} \quad k = i_j \end{cases}$$

By definition, $i_j < j$; therefore, $[R^n]$ is a block lower triangular matrix. Here N_g is the total number of generalized coordinates in the system, i.e.,

$$\sum_{i=1}^{N} N_i$$

The matrix $[U - R^n]$ is lower triangular having 1 on all of the diagonal elements. Therefore, it has full rank and, hence, is invertible.

E. Alternate Velocity Transformation

The kinematic expression in Eq. (9) is useful for deriving an order N algorithm to invert the mass matrix. However, if the same relation is used for the derivation of the centrifugal, Coriolis, and gravitational terms, the computational advantage of the algorithm will be lost. Therefore, alternate expression for the velocity transformation is proposed. Using Eq. (1), $\{\dot{\boldsymbol{q}}^t_j\}$ can be expressed recursively in terms of $\{\dot{\boldsymbol{q}}_k\}$ as follows:

$$\left\{\dot{\boldsymbol{q}}^t_1\right\} = [R_1]\{\dot{\boldsymbol{q}}_1\} \tag{10a}$$

$$\left\{\dot{\boldsymbol{q}}^t_j\right\} = [R_j]\{\dot{\boldsymbol{q}}_j\} + \sum_{k \in S^i_j} \left[R^v_{jk}\right]\{\dot{\boldsymbol{q}}_k\}, \qquad j = 2, \ldots, N \tag{10b}$$

where

$$\left[R^v_{jk}\right] = \begin{bmatrix} \left[T_{i_k}\right] & \left[P_k\left(d_{o_{kj}}\right)\right] & [T_k]\left[\Phi_k\left(d^r_{o_{kj}}\right)\right] \\ O & O & O \end{bmatrix} \in \Re^{N_j \times N_k}$$

and $[R_j]$ is defined in Eq. (8). Arranging for all of the bodies, the preceding equation becomes

$$\{\dot{\boldsymbol{q}}^t\} = [R^v]\{\dot{\boldsymbol{q}}\} \tag{11}$$

where for $j = 1, \ldots, N$ and $k = 1, \ldots, N$, $[R^v]$ is defined by

$$(j, k)\text{th block of } [R^v] = \begin{cases} [R_j], & \text{if} \quad j = k \\ \left[R^v_{jk}\right], & \text{if} \quad k \in S^i_j \\ O, & \text{if} \quad k \notin S_j \end{cases}$$

By definition, $k < j, \forall k \in S^i_j$; therefore, $[R^v]$ is a lower block triangular matrix. For a given $k < l$ and $k < r$, if $o_{kl} = o_{kr}$, then $[R^v_{lk}] = [R^v_{rk}]$.

F. Kinematic Relations

The partial derivatives of the transformed velocity vector $\{\dot{\boldsymbol{q}}^t\}$ with respect to the body fixed velocity vector $\{\dot{\boldsymbol{q}}\}$ and coordinate vector $\{\boldsymbol{q}\}$ are required in deriving the present algorithm. Let the partial derivative of a vector $\{\boldsymbol{x}(y)\} \in \Re^n$ with respect to the vector $\{\boldsymbol{y}\} \in \Re^n$ be defined by the matrix

$$\left[\frac{\partial x}{\partial y}\right] = \left[\left\{\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{y}_1}\right\} \;\middle|\; \left\{\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{y}_2}\right\} \;\middle|\; \cdots \;\middle|\; \left\{\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{y}_n}\right\}\right]$$

where

$$\left\{\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{y}_j}\right\} = \begin{Bmatrix} \dfrac{\partial \boldsymbol{x}_1}{\partial \boldsymbol{y}_j} \\ \dfrac{\partial \boldsymbol{x}_2}{\partial \boldsymbol{y}_j} \\ \vdots \\ \dfrac{\partial \boldsymbol{x}_n}{\partial \boldsymbol{y}_j} \end{Bmatrix}$$

and x_j and y_j, $j = 1, \ldots, n$, are the elements of vectors $\{\boldsymbol{x}\}$ and $\{\boldsymbol{y}\}$, respectively. Using this definition, Eqs. (9) and (11) yield

$$\left[\frac{\partial \dot{\boldsymbol{q}}^t}{\partial \dot{\boldsymbol{q}}}\right] = [U - R^n]^{-1}[R] \tag{12a}$$

$$\left[\frac{\partial \dot{\boldsymbol{q}}^t}{\partial \dot{\boldsymbol{q}}}\right] = [R^v] \tag{12b}$$

Because $\{\boldsymbol{q}^t\} = \{\boldsymbol{q}^t(q, t)\}$, it can be shown that[21]

$$\left[\frac{\partial q^t}{\partial q}\right] = \left[\frac{\partial \dot{q}^t}{\partial \dot{q}}\right] \qquad \text{and} \qquad \left[\frac{\partial \dot{q}^t}{\partial q}\right] = \frac{\mathrm{d}}{\mathrm{d}t}\left[\frac{\partial \dot{q}^t}{\partial \dot{q}}\right] \tag{13}$$

III. Lagrangian Order *N* Formulation

A. Computation of Energy

The objective here is to get the energy expressions as functions of the transformed generalized coordinates. To that end, the position vector $\{\boldsymbol{R}_{\mathrm{d}m_j}\}$ of an elemental mass $\mathrm{d}m_j$ in B_j with respect to the inertial frame F_I can be obtained as

$$\left\{\boldsymbol{R}_{\mathrm{d}m_j}\right\} = \{\boldsymbol{D}_j\} + [T_j]\{R_j\} = \{\boldsymbol{D}_j\} + [T_j]\{r_j\} + [T_j]\{\boldsymbol{f}_j\} \tag{14}$$

(Fig. 2), where $\{\boldsymbol{D}_j\}$ is the position vector of the origin of frame F_j with respect to F_I, $\{\boldsymbol{R}_j\} = \{r_j\} + \{f_j\}$ is the position vector of the elemental mass $\mathrm{d}m_j$ with respect to F_j, $\{r_j\}$ is the rigid component of $\{\boldsymbol{R}_j\}$, $\{\boldsymbol{f}_j\}$ is the flexible component of $\{\boldsymbol{R}_j\}$, and $[T_j]$ is the transformation matrix between frames F_j and F_I. Using Eq. (5) and differentiating the preceding equation, the inertial velocity of the elemental mass $\mathrm{d}m_j$ can be obtained as

$$\left\{\dot{\boldsymbol{R}}_{\mathrm{d}m_j}\right\} = \left[[U] \ \vdots \ [P_j(R_j)] \ \vdots \ [T_j][\Phi_j]\right]\left\{\dot{\boldsymbol{q}}^t_j\right\} \tag{15}$$

Now the kinetic energy of B_j can be expressed as

$$K_j = \frac{1}{2}\int_{m_j} \left\{\dot{\boldsymbol{R}}_{\mathrm{d}m_j}\right\} \cdot \left\{\dot{\boldsymbol{R}}_{\mathrm{d}m_j}\right\} \mathrm{d}m_j = \frac{1}{2}\left\{\dot{\boldsymbol{q}}^t_j\right\}\left[M^t_j\right]\left\{\dot{\boldsymbol{q}}^t_j\right\} \tag{16}$$

where $[M^t_j]$ is the mass matrix of B_j corresponding to the transformed coordinates $\{\boldsymbol{q}^t_j\}$.

For an orbiting system, the gravitational potential energy (P_{G_j}) of the jth body can be expressed as

$$P_{G_j} = -\int_{m_j} \frac{GM_e \, \mathrm{d}m_j}{(\{\boldsymbol{D}_j + \boldsymbol{R}_j\} \cdot \{\boldsymbol{D}_j + \boldsymbol{R}_j\})^{\frac{1}{2}}} \tag{17}$$

where G is the universal gravitational constant and M_e the mass of the Earth. For a ground-based system, the potential energy can be obtained using a constant gravitational acceleration.

Almost all flexible structural elements encountered in space or robotics applications can be modeled as beams (robot arms, satellite antennas, space station power boom, etc.), strings (tether), or plates (solar panels). The elastic potential energy associated with the jth body can be expressed as

$$P_{S_j} = \frac{1}{2}\int_{V_j} \sigma_j \epsilon_j \, \mathrm{d}V_j \tag{18}$$

where σ_j is the stress in an elemental volume $\mathrm{d}V_j$, ϵ_j the strain in the element, and V_j the volume of the jth body. As mentioned in Sec. II.C, the nonlinear strain-displacement relation is used to include the geometric stiffening effects. For a beam element, where the Y_j axis is along the length, the nonlinear strain can be expressed as

$$\epsilon_j \approx \frac{\partial f_{y_j}}{\partial y_j} + \frac{1}{2}\left[\left(\frac{\partial f_{x_j}}{\partial y_j}\right)^2 + \left(\frac{\partial f_{z_j}}{\partial y_j}\right)^2\right]$$

The total energy of the entire system can be obtained by adding energies of the individual bodies.

B. Order *N* Dynamical Formulation

The governing equations of motion can be obtained using the classical Lagrangian procedure

$$\frac{\mathrm{d}}{\mathrm{d}t}\left\{\frac{\partial K}{\partial \dot{q}}\right\} - \left\{\frac{\partial (K - P)}{\partial q}\right\} = \{\boldsymbol{Q}_q\} \tag{19}$$

where $P = P_G + P_S$; K, P_G, and P_S are the kinetic, gravitational potential, and strain energies of the entire system, respectively; and $\boldsymbol{Q}_q$ is the generalized force vector due to nonconservative forces such as control inputs, environmental effects, structural damping, etc. The generalized forces can be obtained using the principle of virtual work. Applying the chain rule of differentiation, Eq. (19) can be written as

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\left[\frac{\partial \dot{q}^t}{\partial \dot{q}}\right]^T \left\{\frac{\partial K}{\partial \dot{\boldsymbol{q}}^t}\right\}\right) - \left[\frac{\partial \dot{q}^t}{\partial q}\right]^T \left\{\frac{\partial K}{\partial \dot{\boldsymbol{q}}^t}\right\}$$

$$- \left[\frac{\partial q^t}{\partial q}\right]^T \left\{\frac{\partial (K - P)}{\partial q^t}\right\} = \{\boldsymbol{Q}_q\} \tag{20}$$

Using the kinematics relations of Eqs. (12) and (13), Eq. (20) can be expressed as

$$[M]\{\ddot{\boldsymbol{q}}\} + \{F\} = \{\boldsymbol{Q}_q\} \tag{21}$$

where

$$[M] = [R]^T[U - R^n]^{-T}[M^t][U - R^n]^{-1}[R]$$

$$\{F\} = [R^v]^T\left\{[\dot{M}^t][R^v]\{\dot{\boldsymbol{q}}\} + [M^t][\dot{R}^v]\{\dot{\boldsymbol{q}}\} + \left\{\frac{\partial (K - P)}{\partial q^t}\right\}\right\}$$

and $[M^t]$ is a block diagonal matrix with $[M^t_j]$ as the jth diagonal block [Eq. (16)]. From the preceding equation, the inverse of the mass matrix can be obtained as

$$[M]^{-1} = [R]^{-1}[U - R^n][M^t]^{-1}[U - R^n]^T[R]^{-T} \tag{22}$$

The calculation of $[M]^{-1}$ by directly inverting the matrix $[M]$ requires $\mathcal{O}(N^3)$ computations. However, the use of Eq. (22) for calculating $[M]^{-1}$ needs computation of the inverses of $[M^t]$ and $[R]$, which are block diagonal, and formulation of the matrix $[R^n]$, which depends on system parameters and generalized coordinates. Therefore, the computational effort required for the inversion of the mass matrix using Eq. (22) is proportional to the number of diagonal blocks in $[M^t]$, i.e., the number of bodies N.

Computation of $[R^v]$, $[M^t]$, and their time derivatives used in the expression for $\{F\}$ require order N arithmetic operations. The partial derivatives of the kinetic K and the potential P energies with respect to $\{\boldsymbol{q}^t\}$ also require order N computations. Furthermore, the number of operations required to solve Eq. (21) for $\{\ddot{\boldsymbol{q}}\}$ is order N (Appendix). Therefore, the forward dynamics computation, i.e., obtaining the acceleration vector, requires order N floating point operations.

C. Advantages of the Proposed Method

This algorithm has several advantages compared to other order N methods available in the literature. The features of different approaches are compared in Table 1. Among all of the order N algorithms, the formulation of Menon and Kurdila[7] is the only nonrecursive method. The present algorithm is mostly nonrecursive, except for the computation of the Lagrange multipliers, which allows for its implementation in parallel computers. As opposed to other methods, the Lagrangian approach has two major advantages: 1) the energy expressions are explicitly obtained during the formulation and can be used to verify the equations and 2) the internal constraint forces are automatically taken into consideration. Furthermore, the equations are simple to derive and implement numerically.

In a conventional Lagrangian approach, the complexity of multibody formulation increases as the number of bodies increases. Therefore, use of symbolic manipulation is preferred in Lagrangian

Table 1 Comparison of different order *N* algorithms for studying multibody systems

Formulation	Basic approach	System topology	Dynamics	Recursive or nonrecursive
Bae and Haug[2]	Virtual work	Flexible tree	Forward	Recursive
van Woerkom and de Boer[4]	Newton–Euler	Flexible chain	Forward	Recursive
Keat[5]	Velocity transform	Flexible tree	Forward	Recursive
Rosenthal[6]	Kane's equation	Flexible tree	Forward	Recursive
Menon and Kurdila[7]	Range space method	Flexible tree	Forward	Nonrecursive
Rodriguez and Kreutz-Delgado[8]	Filtering and smoothing	Flexible tree	Forward	Recursive
Hollerbach[1]	Lagrangian	Rigid chain	Inverse	Recursive
Proposed	Lagrangian	Flexible tree	Forward	Mostly nonrecursive

formulation.[24] One of the advantages of the current approach is that the complexity of the algorithm does not increase as the number of bodies increases. To get the equations of motion, one has to derive equations for a single flexible body in space and the transformation matrices. Therefore, use of symbolic manipulation is not a major consideration in the present study. However, in conjunction with the proposed algorithm, it may be helpful to use symbolic manipulation to get closed-form expressions for the modal integral matrices. This will eliminate the numerical computation of these matrices.

The only available Lagrangian order N formulation by Hollerbach[1] develops an inverse dynamics algorithm for a rigid multibody system in chain topology. If used for the forward dynamics study, the algorithm will require order N^3 computations. The current formulation considers the most general system configuration, i.e., flexible bodies connected in a tree topology including closed loops. This is the only available order N forward dynamics algorithm based on the Lagrangian approach.

D. Systems with Constrained Generalized Coordinates

For a system with closed loops, cut joints are introduced into the system configuration to make it an open tree. This adds certain constraints, which can be included in the formulation by the Lagrange multipliers. The constraints for all of the cut joints can be expressed as

$$[C^c]^T\{\dot{q}\} = 0 \tag{23}$$

where $[C^c]$ is the coefficient matrix for cut joint constraints. Furthermore, the algorithm requires $\{\boldsymbol{d}^r_j\}$, $j = 1, \dots, N$, to be generalized coordinates. But in many practical situations, where translational degrees of freedom are absent, $\{\boldsymbol{d}^r_j\}$ will be constant. As in the case of cut joints, these constraints are also incorporated by the Lagrange multipliers.

Let c_k, $k = 1, \dots, N^u$, be the number of the bodies having no translational degrees of freedom. Here N^u is the total number of bodies having the constraints. Here c_k is arranged in such a way that for any two integers r and s between 1 and N^u, if $r > s$, then $c_r > c_s$. Using these notations, the constraint equations can be expressed as

$$\left[C^u_l\right]^T\{\dot{\boldsymbol{q}}\} = 0, \qquad l = 1, \dots, N^u \tag{24}$$

The coefficient matrices $[C^u_l]$, $l = 1, \dots, N^u$ for $r = 1, \dots, N^u$, are defined as

$$r\text{th block of } \left[C^u_l\right] = \begin{cases} O, & \text{if} \quad r \neq c_l \\ [U_r], & \text{if} \quad r = c_l \end{cases}$$

where

$$[U_r] = \begin{bmatrix} [U] \\ O \end{bmatrix} \in \Re^{N_r \times 3}$$

The governing equations of motion of the constrained system become[21]

$$\{\ddot{\boldsymbol{q}}\} = \left\{\tilde{F}_{N^u}\right\} + \sum_{k=1}^{N^u} \left[\tilde{C}^u_{N^u k}\right]\left\{\Lambda^u_k\right\} + \left[\tilde{C}^c_{N^u}\right]\{\Lambda^c\} \tag{25}$$

where

$$\left\{\tilde{F}_{N^u}\right\} = [M]^{-1}\{-F + Q_q\}$$

$$\left[\tilde{C}^u_{N^u k}\right] = [M]^{-1}\left[C^u_k\right], \qquad k = 1, \dots, N^u$$

$$\left[\tilde{C}^c_{N^u}\right] = [M]^{-1}[C^c]$$

Here $\{\Lambda^u_k\}$, $k = 1, \dots, N^u$, and $\{\Lambda^c\}$ are the Lagrange multipliers for constraints at uncut and cut joints, respectively. Equations (24) and (25) can be solved recursively for the Lagrange multipliers for uncut joints for $j = N^u, \dots, 1$, as given by the following recursive algorithm:

$$\left\{\Lambda^u_j\right\} = \left\{L^1_j\right\} + \sum_{k=1}^{j-1} \left[L^2_{jk}\right]\left\{\Lambda^u_k\right\} + \left[L^3_j\right]\{\Lambda^c\} \tag{26}$$

$$\{\ddot{\boldsymbol{q}}\} = \{\tilde{F}_{j-1}\} + \sum_{k=1}^{j-1} \left[\tilde{C}^u_{(j-1)k}\right]\left\{\Lambda^u_k\right\} + \left[\tilde{C}^c_{j-1}\right]\{\Lambda^c\} \tag{27}$$

where

$$\left\{L^1_j\right\} = \left(\left[C^u_j\right]^T\left[\tilde{C}^u_{jj}\right]\right)^{-1}\left[C^u_j\right]^T\{\tilde{F}_j\}$$

$$\left[L^2_{jk}\right] = \left(\left[C^u_j\right]^T\left[\tilde{C}^u_{jj}\right]\right)^{-1}\left[C^u_j\right]^T\left[\tilde{C}^u_{jk}\right], \qquad k = 1, \dots, (j-1)$$

$$\left[L^3_j\right] = \left(\left[C^u_j\right]^T\left[\tilde{C}^u_{jj}\right]\right)^{-1}\left[C^u_j\right]^T\left[\tilde{C}^c_j\right]$$

$$\{\tilde{F}_{j-1}\} = \{\tilde{F}_j\} + \left[\tilde{C}^u_{jj}\right]\left\{L^1_j\right\}$$

$$\left[\tilde{C}^u_{(j-1)k}\right] = \left[\tilde{C}^u_{jk}\right] + \left[\tilde{C}^u_{jj}\right]\left[L^2_{jk}\right], \qquad k = 1, \dots, (j-1)$$

$$\left[\tilde{C}^c_{j-1}\right] = \left[\tilde{C}^c_j\right] + \left[\tilde{C}^u_{jj}\right]\left[L^3_j\right]$$

For $j = 1$, Eq. (27) contains only the Lagrange multipliers at cut joints. The resulting equation is

$$\{\ddot{\boldsymbol{q}}\} = \{\tilde{F}_0\} + \left[\tilde{C}^c_0\right]\{\Lambda^c\} \tag{28}$$

Using Eqs. (23) and (28),

$$\{\Lambda^c\} = \left([C^c]^T\left[\tilde{C}^c_0\right]\right)^{-1}[C^c]^T\{\tilde{F}_0\} \tag{29}$$

The number of algebraic operations required to solve $\{\Lambda^u_k\}$, $k = 1, \dots, N^u$, depends on the structure of the system, mainly the number of chains in the graph G' (Fig. 1). For a system in chain topology, it is exactly order N. Solving for $\{\Lambda^c\}$ requires order $(N^c)^3$ operations, where N^c is the number of constraints at all of the cut joints. In most systems, this number is small compared to the degrees of freedom and, therefore, does not affect the efficiency of the formulation.

IV. Illustrative Example

A sample problem is presented to show details of the steps involved in deriving the equations of motion using the proposed algorithm. The example considers the space-station-based mobile remote manipulator system (MRMS). The system consists of a two-link manipulator (B_2, B_3) capable of translation on the space platform B_1 (Fig. 3). The end effector consists of two smaller two-link robots, for dexterous operations, and is attached to B_3. The bodies are numbered according to the scheme defined earlier. The bodies

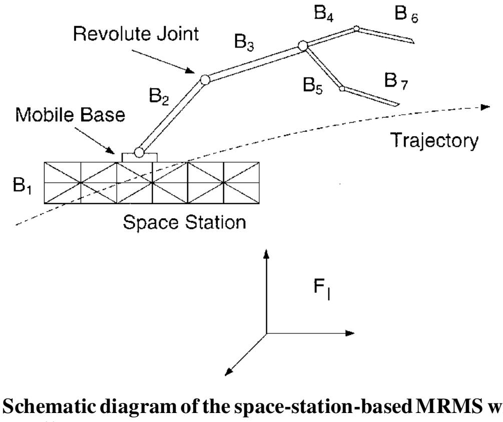


Fig. 3 Schematic diagram of the space-station-based MRMS with dexterous end effector.

B_k, $k = 3, 4, \ldots, 7$, have constraints of type $\dot{d}^r_k = 0$. The transformation matrices for this system can be expressed as

$$[R] = \begin{bmatrix} [R_1] & & & & & & \\ & [R_2] & & O & & & \\ & & [R_3] & & & & \\ & & & [R_4] & & & \\ & & & & [R_5] & & \\ & & O & & & [R_6] & \\ & & & & & & [R_7] \end{bmatrix}$$

$$[R^n] = \begin{bmatrix} O & & & & & & \\ \left[R^n_{21}\right] & O & & & & & \\ O & \left[R^n_{32}\right] & O & & O & & \\ O & O & \left[R^n_{43}\right] & O & & & \\ O & O & \left[R^n_{53}\right] & O & O & & \\ O & O & O & \left[R^n_{64}\right] & O & O & \\ O & O & O & O & \left[R^n_{75}\right] & O & O \end{bmatrix}$$

$$[R^v] = \begin{bmatrix} [R_1] & & & & & & \\ \left[R^v_{21}\right] & [R_2] & & & & & \\ \left[R^v_{31}\right] & \left[R^n_{32}\right] & [R_3] & & O & & \\ \left[R^v_{41}\right] & \left[R^v_{42}\right] & \left[R^n_{43}\right] & [R_4] & & & \\ \left[R^v_{51}\right] & \left[R^v_{52}\right] & \left[R^n_{53}\right] & O & [R_5] & & \\ \left[R^v_{61}\right] & \left[R^v_{62}\right] & \left[R^n_{63}\right] & \left[R^n_{64}\right] & O & [R_6] & \\ \left[R^v_{71}\right] & \left[R^v_{72}\right] & \left[R^n_{73}\right] & O & \left[R^n_{75}\right] & O & [R_7] \end{bmatrix}$$

The submatrices of $[R]$, $[R^n]$, and $[R^v]$ are defined as discussed in the preceding section. Following the procedure outlined before, the mass matrix of the jth body corresponding to the transformed coordinates [Eq. 16)] can be expressed as

$$\left[M^t_j\right] = \begin{bmatrix} \left[M^t_{D_j}\right] & \left[M^t_{D\eta_j}\right] & \left[M^t_{D\delta_j}\right] \\ \left[M^t_{D\eta_j}\right]^T & \left[M^t_{\eta_j}\right] & \left[M^t_{\eta\delta_j}\right] \\ \left[M^t_{D\delta_j}\right]^T & \left[M^t_{\eta\delta_j}\right]^T & \left[M^t_{\delta_j}\right] \end{bmatrix} \in \Re^{N_j \times N_j} \tag{30}$$

where

$$\left[M^t_{D_j}\right] = m_j[U] \in \Re^{3\times 3}$$

$$\left[M^t_{D\eta_j}\right] = \int_{m_j} [P_j(R_j)]\,\mathrm{d}m_j \in \Re^{3\times 3}$$

$$\left[M^t_{D\delta_j}\right] = [T_j]\int_{m_j} [\Phi_j]\,\mathrm{d}m_j \in \Re^{3\times n_j}$$

$$\left[M^t_{\eta_j}\right] = \int_{m_j} [P_j(R_j)]^T[P_j(R_j)]\,\mathrm{d}m_j \in \Re^{3\times 3}$$

$$\left[M^t_{\eta\delta_j}\right] = \int_{m_j} [P_j(R_j)]^T[T_j][\Phi_j]\,\mathrm{d}m_j \in \Re^{3\times n_j}$$

$$\left[M^t_{\delta_j}\right] = \int_{m_j} [\Phi_j]^T[\Phi_j]\,\mathrm{d}m_j \in \Re^{n_j\times n_j}$$

and the subscripts D, η, and δ are associated with the generalized coordinates $\{\boldsymbol{D}_j\}$, the attitude angles $\{\eta_j\}$, and the flexible generalized coordinates $\{\delta_j\}$, respectively. $[\Phi_j]$ is defined in Eq. (5), whereas Eq. (8) defines $[T_j]$ and $[P(R_j)]$. The time derivative of the mass matrix is obtained by differentiating Eq. (30).

To obtain the equations of motion using Eq. (21), it is required to calculate the partial derivative of the kinetic energy K with respect to the generalized coordinate vector $\{\boldsymbol{q}^t\}$. Because K_j is a function of only $\{\boldsymbol{q}^t_j\}$, the differentiation leads to the following:

$$\left\{\frac{\partial K_i}{\partial \boldsymbol{q}^t_j}\right\} = \{0\}, \qquad \forall i \neq j$$

$$\left\{\frac{\partial K_j}{\partial \boldsymbol{q}^t_j}\right\} = \begin{Bmatrix} \left\{\dfrac{\partial K_j}{\partial D_j}\right\} \\ \left\{\dfrac{\partial K_j}{\partial \eta_j}\right\} \\ \left\{\dfrac{\partial K_j}{\partial \delta_j}\right\} \end{Bmatrix} \tag{31}$$

Because K_j is not a function of $\{\boldsymbol{D}_j\}$, $\{\partial K_j/\partial \boldsymbol{D}_j\} = \{0\}$. Because of the trigonometric terms in $[M^t_j]$, differentiation of K_j with respect to $\{\eta_j\}$ must account for the individual angles, i.e.,

$$\left\{\frac{\partial K_j}{\partial \eta_j}\right\} = \frac{1}{2}\begin{Bmatrix} \left\{\dot{\boldsymbol{q}}^t_j\right\}^T \left[\dfrac{\partial M^t_j}{\partial \alpha_j}\right]\left\{\dot{\boldsymbol{q}}^t_j\right\} \\ \left\{\dot{\boldsymbol{q}}^t_j\right\}^T \left[\dfrac{\partial M^t_j}{\partial \beta_j}\right]\left\{\dot{\boldsymbol{q}}^t_j\right\} \\ \left\{\dot{\boldsymbol{q}}^t_j\right\}^T \left[\dfrac{\partial M^t_j}{\partial \gamma_j}\right]\left\{\dot{\boldsymbol{q}}^t_j\right\} \end{Bmatrix} \tag{32}$$

where

$$\left[\frac{\partial M^t_j}{\partial \alpha_j}\right] = \begin{bmatrix} \left[\dfrac{\partial M^t_{D_j}}{\partial \alpha_j}\right] & \left[\dfrac{\partial M^t_{D\eta_j}}{\partial \alpha_j}\right] & \left[\dfrac{\partial M^t_{D\delta_j}}{\partial \alpha_j}\right] \\ & \left[\dfrac{\partial M^t_{\eta_j}}{\partial \alpha_j}\right] & \left[\dfrac{\partial M^t_{\eta\delta_j}}{\partial \alpha_j}\right] \\ \text{Sym.} & & \left[\dfrac{\partial M^t_{\delta_j}}{\partial \alpha_j}\right] \end{bmatrix}$$

$$\left[\frac{\partial M^t_{D_j}}{\partial \alpha_j}\right] = [O], \qquad \left[\frac{\partial M^t_{D\delta_j}}{\partial \alpha_j}\right] = \left[\frac{\partial T_j}{\partial \alpha_j}\right]\int_{m_j} [\Phi_j]\,\mathrm{d}m_j$$

and so on. The differentiation of K_j with respect to the flexible generalized coordinates can be simplified by expressing the kinetic energy of the jth body as

$$K_j = \tfrac{1}{2}\left\{\dot{\boldsymbol{q}}^t_j\right\}^T[\bar{M}_j]\left\{\dot{\boldsymbol{q}}^t_j\right\} + \tfrac{1}{2}\{\delta_j\}^T[\bar{K}_j]\{\delta_j\} + \{\delta_j\}^T[\bar{G}_j]\left\{\dot{\boldsymbol{q}}^t_j\right\}$$

where $[\bar{M}_j]$, $[\bar{K}_j]$, and $[\bar{G}_j]$ do not depend on $\{\delta_j\}$. Applying simple algebraic manipulations, it can be shown that

$$\left\{\frac{\partial K_j}{\partial \delta_j}\right\} = [\bar{K}_j]\{\delta_j\} + [\bar{G}_j]\left\{\dot{\boldsymbol{q}}^t_j\right\} \tag{33}$$

where

$$[\bar{K}_j] = \int_{m_j} [\Phi_j]^T[\dot{T}_j]^T[\dot{T}_j][\Phi_j]\,\mathrm{d}m_j \in \Re^{n_j\times n_j}$$

$$[\bar{G}_j] = \left[\left[\bar{G}_{D_j}\right] \quad \left[\bar{G}_{\eta_j}\right] \quad \left[\bar{G}_{\delta_j}\right]\right]$$

$$\left[\bar{G}_{D_j}\right] = \left[\int_{m_j} [\Phi_j]^T\,\mathrm{d}m_j\right][\dot{T}_j]^T \in \Re^{n_j\times 3}$$

$$\left[\bar{G}_{\eta_j}\right] = \int_{m_j} [\Phi_j]^T[\dot{T}_j]^T[P_j(r_j)]\,\mathrm{d}m_j \in \Re^{n_j\times 3}$$

$$\left[\bar{G}_{\delta_j}\right] = \int_{m_j} [\Phi_j]^T[\dot{T}_j]^T[T_j][\Phi_j]\,\mathrm{d}m_j \in \Re^{n_j\times n_j}$$

The differentiation of the potential energy with respect to the generalized coordinates is obtained in a similar way, and the governing equations of motion can be expressed as in Eq. (21). The details of the equations are not reported for brevity. Here, $\{\boldsymbol{d}_1^r\}$ describes the orbital motion of the system and $\{\boldsymbol{d}_2^r\}$ represents the motion of the mobile base of the MRMS. All other joints have only rotational degrees of freedom. Therefore, $\{\boldsymbol{d}_j^r\}$, $j = 3, \ldots, 7$, are constant. These nonholonomic constraint relations are introduced into the equations of motion using the Lagrange multipliers [Eqs. (26–29)].

V. Concluding Remarks

A new Lagrangian order N algorithm is developed for the dynamical formulation of flexible multibody systems in tree topology including closed loops. Application of different forms of velocity transformation in conjunction with the chain rule of differentiation, leads to a factorization of the mass matrix, which results in an order N forward dynamics algorithm. The constrained degrees of freedom are introduced by the Lagrange multipliers, which can be solved recursively. The space-station-based MRMS with a dexterous end effector, considered as an example, helps explain the procedure involved.

This novel algorithm aimed at the Lagrangian order N formulation for forward dynamics of flexible systems in tree topology is, indeed, versatile. The algorithm is mostly nonrecursive, which facilitates its implementation in parallel computers. The use of the Lagrangian approach gives explicit energy expressions, which can be used for verifying the formulation. The algorithm is particularly attractive for its simplicity in derivation and implementation.

Appendix: Order *N* Solution of $\{\ddot{\boldsymbol{q}}\}$

Computation of the vector $\{\ddot{\boldsymbol{q}}\}$ by solving Eq. (21) requires multiplication of the matrix $[R^v]$ with the vector $\{\dot{\boldsymbol{q}}\}$. Using results from the graph theory, it can be shown that this multiplication is an order N operation.

The number of different nonzero blocks in an arbitrary kth column of $[R^v]$ is $(N_k^o + 1)$, where N_k^o is the number of outboard bodies of B_k. Let the system be represented by a digraph (directed graph),[12] where the edge between vertices B_j and B_{i_j} has its tail at B_{i_j} and head at B_j. Then N_j^o is the out degree of the vertex B_j $[od(B_k)]$. Therefore, to multiply the kth block column of $[R^v]$ with $\{\dot{\boldsymbol{q}}_k\}$, it is necessary to multiply $(N_k^o + 1)$ submatrices with vectors. Then, the total number of submatrix multiplication in $[R^v]\{\dot{\boldsymbol{q}}\}$ is

$$\sum_{k=1}^{N} [od(B_k) + 1]$$

Using results from the graph theory,[12] it can be shown that

$$\sum_{k=1}^{N} [od(B_k) + 1] = 2N - 1$$

Thus, the total number of operation required to get $[R^v]\{\dot{\boldsymbol{q}}\}$ is order N.

References

[1] Hollerbach, J. M., "A Recursive Lagrangian Formulation of Manipulator Dynamics and a Comparative Study of Dynamics Formulation Complexity," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-10, No. 11, 1980, pp. 730–736.

[2] Bae, D. S., and Haug, E. J., "A Recursive Formulation for Constrained Mechanical System Dynamics: Part. I, Open Loop Systems," *Mechanics of Structures and Machines*, Vol. 15, No. 3, 1987, pp. 359–382.

[3] Roberson, R. E., and Schwertassek, R., *Dynamics of Multibody Systems*, Springer–Verlag, Berlin, 1988, Chaps. 8–14.

[4] Van Woerkom, P. T. L. M., and de Boer, A., "Development and Validation of a Linear Recursive 'Order-N' Algorithm for the Simulation of Flexible Space Manipulator Dynamics," *Acta Astronautica*, Vol. 35, Nos. 2/3, 1995, pp. 175–185.

[5] Keat, J. E., "Multibody System Order *N* Dynamics Formulation Based on Velocity Transform Method," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 2, 1990, pp. 207–212.

[6] Rosenthal, D. E., "An Order *N* Formulation for Robotic Systems," *Journal of Astronautical Sciences*, Vol. 38, No. 4, 1990, pp. 511–529.

[7] Menon, R. G., and Kurdila, A. J., "An Alternative Order *N* Formulation for Simulation and Control of Multibody Systems," *Proceedings of the AIAA Guidance, Navigation, and Control Conference* (Hilton Head Island, SC), AIAA, Washington, DC, 1992, pp. 134–142.

[8] Rodriguez, G., and Kreutz-Delgado, K., "Spatial Operator Factorization and Inversion of the Manipulator Mass Matrix," *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 1, 1992, pp. 65–76.

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

[10] Ho, J. Y. L., and Herber, D. R., "Development of Dynamics and Control Simulation of Large Flexible Space Systems," *Journal of Guidance, Control, and Dynamics*, Vol. 8, No. 3, 1985, pp. 374–383.

[11] Singh, R. P., VanderVoort, R. J., and Likins, P. W., "Dynamics of Flexible Bodies in Tree Topology—A Computer-Oriented Approach," *Journal of Guidance, Control, and Dynamics*, Vol. 8, No. 5, 1985, pp. 584–590.

[12] Clark, J., and Holton, D. A., *A First Look at Graph Theory*, World Scientific, Singapore, 1991, p. 238.

[13] Hughes, T. J. R., *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Prentice–Hall, Englewood Cliffs, NJ, 1987, pp. 1–13.

[14] Meirovitch, L., *Dynamics and Control of Structures*, Wiley, New York, 1989, pp. 283–309.

[15] Hurty, W. C., "Dynamic Analysis of Structural Systems Using Component Modes," *AIAA Journal*, Vol. 3, No. 4, 1965, pp. 678–685.

[16] Hughes, P. C., "Space Structure Vibration Modes: How Many Exist? Which Ones Are Important?" *IEEE Control Systems Magazine*, Vol. 7, No. 1, 1987, pp. 22–28.

[17] Kane, T. R., Ryan, R. R., and Banerjee, A. K., "Dynamics of Cantilever Beam Attached to a Moving Base," *Journal of Guidance, Control, and Dynamics*, Vol. 10, No. 2, 1987, pp. 139–151.

[18] Sharf, I., "Geometric Stiffening in Multibody Dynamics Formulations," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 4, 1995, pp. 882–890.

[19] Hanagud, S., and Sarkar, S., "Problem of the Dynamics of a Cantilever Beam Attached to a Moving Base," *Journal of Guidance, Control, and Dynamics*, Vol. 12, No. 3, 1989, pp. 438–441.

[20] Pestel, J. E. C., and Leckie, F. A., *Matrix Methods in Elastomechanics*, McGraw–Hill, New York, 1963, Chap. 4.

[21] Greenwood, D. T., *Principles of Dynamics*, 2nd ed., Prentice–Hall, Englewood Cliffs, NJ, 1988, pp. 265, 523.

[22] Book, W. J., "Structural Flexibility of Motion Systems in the Space Environment," *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 5, 1993, pp. 524–530.

[23] Jerkovsky, W., "The Structure of Multibody Dynamics Equations," *Journal of Guidance and Control*, Vol. 1, No. 3, 1978, pp. 173–182.

[24] Keshmiri, M., and Misra, A. K., "A General Formulation for N-Body Tethered Satellite System Dynamics," *Advances in Astronautical Sciences*, edited by A. K. Misra, V. J. Modi, R. Holdaway, and P. M. Bainum, Vol. 85, Pt. 1, Univelt, San Deigo, CA, 1994, pp. 623–644.