Multibody Syst Dyn (2009) 21: 1–35
DOI 10.1007/s11044-008-9122-6

A recursive, numerically stable, and efficient simulation algorithm for serial robots with flexible links

Ashish Mohan · S.K. Saha

Received: 12 July 2007 / Accepted: 22 July 2008 / Published online: 21 September 2008

Abstract A methodology for the formulation of dynamic equations of motion of a serial flexible-link manipulator using the decoupled natural orthogonal complement (DeNOC) matrices, introduced elsewhere for rigid bodies, is presented in this paper. First, the Euler Lagrange (EL) equations of motion of the system are written. Then using the equivalence of EL and Newton–Euler (NE) equations, and the DeNOC matrices associated with the velocity constraints of the connecting bodies, the analytical and recursive expressions for the matrices and vectors appearing in the independent dynamic equations of motion are obtained. The analytical expressions allow one to obtain a recursive forward dynamics algorithm not only for rigid body manipulators, as reported earlier, but also for the flexible body manipulators. The proposed simulation algorithm for the flexible link robots is shown to be computationally more efficient and numerically more stable than other algorithms present in the literature. Simulations, using the proposed algorithm, for a two link arm with each link flexible and a Space Shuttle Remote Manipulator System (SSRMS) are presented. Numerical stability aspects of the algorithms are investigated using various criteria, namely, the zero eigenvalue phenomenon, energy drift method, etc. Numerical example of a SSRMS is taken up to show the efficiency and stability of the proposed algorithm. Physical interpretations of many terms associated with dynamic equations of flexible links, namely, the mass matrix of a composite flexible body, inertia wrench of a flexible link, etc. are also presented.

Keywords Flexible · DeNOC matrices · Recursive · Simulation · Numerical stabile · SSRMS

The work has been carried out in the Dept. of Mechanical Engineering, Indian Institute of Technology Delhi, New Delhi 110016, India.

A. Mohan
Hi-Tech Robotic Systemz Limited, Gurgaon, India
e-mail: ashishsept13@rediffmail.com

S.K. Saha (✉)
Department of Mechanical Engineering, Indian Institute of Technology Delhi, New Delhi, India
e-mail: saha@mech.iitd.ernet.in

1 Introduction

The necessity and importance of dynamic modeling of multibody systems with structurally flexible links is unambiguous. Research on the robotic systems with flexible arms and its control started in the international arena in early 1970s. A comprehensive review of the various techniques on the modeling of robot-link flexibility is given in [7, 42, 43, 50, 53]. A number of efficient dynamic algorithms for flexible multibody systems, based on different approaches, are present in the literature [3, 9, 11, 15, 17, 20]. Most of the algorithms recognize the difficulty of expressing the dynamic equations for the flexible links in Newton–Euler (NE) form. On the other hand, the equations can be readily expressed in Euler–Lagrange (EL) form. Thus, although recursive algorithm based on the NE form of dynamic equations for flexible multibody systems have been proposed in the literature, e.g., in [41], researchers tend to prefer the readily available EL form of the dynamic model [4, 27–29, 34, 51]. One of the approaches proposed in the literature is to exploit the advantages of both EL and NE formulations. The methodology is based on the expression of dynamic equations of motion for individual flexible links first in its readily available EL form. Then using the equivalence of EL and NE formulations, the dynamic model for each flexible link is written in the form of its NE equations. Next, the constraint moments and forces are eliminated to obtain the independent set of dynamic equations of the system at hand by projecting the uncoupled equations of motion of all the flexible links on the constrained manifold, as done in [31] for the rigid-link systems.

For the purpose of elimination of the constraint equations, various methods have been proposed in the literature. One such method is based on the use of orthogonal complement of the velocity constraints of mechanical system under study. An orthogonal complement is defined as the matrix whose columns span the null-space of the matrix of the velocity constraints, and hence the premultiplication of its transpose with the unconstrained dynamic equations of motion vanishes the constrained moments and forces. As a result, a set of independent dynamic equations of motion, which are Ordinary Differential Equations (ODE), is obtained. The ODEs are known to provide numerically stable algorithms compared to the Differential Algebraic Equations (DAE) representing the same system dynamics [33]. The said orthogonal complement is, however, not unique. In some approaches, an orthogonal complement is found using numerical schemes, which are of an intensive nature requiring, for example, singular value decomposition or eigenvalue computations [25, 54]. Angeles and Lee [1] and Saha and Angeles [39] have obtained the complement for the serial rigid multibody systems naturally from the velocity constraint expressions without any complex computations. Therefore, the matrix is called the natural orthogonal complement (NOC). Cyril [13] has obtained the above compliment for the flexible serial multibody systems and reported a forward dynamics algorithm, which is not recursive. Saha [35] extended the concept of the NOC by decoupling its representation, called the decoupled natural orthogonal complement (DeNOC) matrices, and has eventually obtained independent dynamic equations from which both recursive inverse and forward dynamics algorithms for serial rigid robotic systems have been obtained [36]. Note that the recursive forward dynamics algorithm was not possible with the original form of the NOC, as reflected in [1, 13]. In this present work, the DeNOC has been extended to the serial flexible multibody systems. The NOC matrix for the flexible serial manipulators [13], is decoupled and expressed as a product of two matrices, of which one is a block triangular and the other is a block diagonal.

The advantages of the DeNOC based methodology for flexible serial manipulators are as follows:

(i) Unlike the NOC, its modified form, i.e., the DeNOC, allows one to write the expressions of the elements of the matrices in analytical recursive form.
(ii) The expressions of the elements for the matrices and vectors associated with the dynamic equations of motion can be obtained in analytical and recursive form, besides allowing for a recursive forward dynamics algorithm which is known to provide stable and more accurate simulation results [7, 35–37].
(iii) The DeNOC approach is built upon the basic mechanics and linear algebra theory, which are easy to apprehend.
(iv) Moreover, physical interpretations of many terms, e.g., the mass matrix of a composite flexible body, etc. are possible, similar to those with the rigid body systems [31, 36].

Note that the simulation is a two step process: (1) forward dynamics, i.e., the computation of the accelerations for the generalized coordinates from the equations of motion of a system at hand for the given actuator forces and torques; and (2) numerical integration of the accelerations computed in step (1) to obtain the corresponding rates and positions. Thus, the nature of simulation results depends as much on the forward dynamics algorithm as on the integrator used in step (2). In fact, it is possible that the simulation results are unstable even when the real system is stable. This could be due to ill-conditioning of the problem. A problem is said to be ill-conditioned if even small changes in the data have the potential to induce large changes in the solution of the problem. A forward dynamics algorithm is numerically stable if it does not introduce any additional sensitivity than that already inherent in the problem due to its physical characteristics, namely, the geometry and mass and inertia properties, etc. An initial approach to reduce the numerical instability of the algorithms and overcome the instability inherent with the numerical integration by controlling the accumulation of errors is given by Baumgarte [6]. In the Baumgarte stabilization [6], an artificial feedback, namely, position and velocity terms are added in the second derivative of the constraint equation of the system. The disadvantage of this method is that there is no reliable method for selecting the intensity of artificial feedback, i.e., the coefficients of the position and velocity terms and an improper selection of these coefficients can lead to erroneous results. Chang and Nikravesh [10] proposed different methods for selecting these coefficients for the artificial feedback to the constraint differential equations. Moreover, it should be noted that Baumgarte stabilization does not solve all possible instabilities, such as those arising due to near kinematic singular configurations [19]. This aspect works in favor of another stabilization method, namely, augmented Lagrangian formulation [33]. A brief overview of various stabilization methods is given in [32, 33]. Different researchers have used different criteria for investigating the numerical stability aspects of the simulation algorithms. Ider [21] has used the Jacobian matrix and investigated its rank for the conditions close to singularity. Sharf and Damaren [45] have taken the example of a Canadarm robot with flexible links and compared four different models to investigate their simulation characteristics. They studied the drift in the energy of the system to test the numerical stability of the algorithms. Ellis et al. [18] have studied the numerical stability of the forward dynamics algorithms using methods based on energy conservation. Similarly, formulation stiffness phenomenon has been used by researchers for investigating the numerical stability aspects [2, 5, 12]. Jain and Rodriguez [23, 24] have focused on the computation of the sensitivity of the mass matrix and developed an analytical expression for the same using spatial operator algebra. Stability aspects of a mobile robot, based on the posture velocity error dynamics are proposed by Shim and Sung [46]. The emphasis is, however, more on

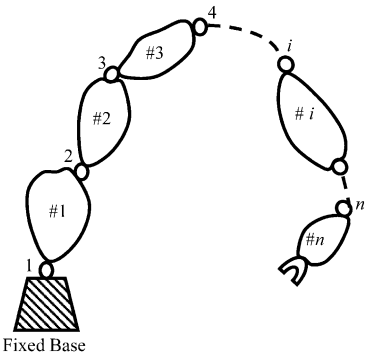


Fig. 1 A typical n-link serial robot

the geometric stability of the systems and numerical stability aspects of dynamic algorithms is not covered.

It will be noted that most of the study on stability of numerical algorithms is limited to the rigid multibody systems. The effect of flexibility of links on the numerical aspects of forward dynamics algorithms still remains a relatively less explored area. In the present work, a recursive, computationally efficient and numerically stable forward dynamics algorithm for serial flexible robots is proposed. The algorithm is based on the $\boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^{\mathrm{T}}$ decomposition of the generalized inertia matrix (GIM) associated with the system's dynamic equations of motion, where $\boldsymbol{U}$ and $\boldsymbol{D}$ are, respectively, the upper block triangular and block diagonal matrices. Analytical expressions for the elements of the matrices, $\boldsymbol{U}$ and $\boldsymbol{D}$, are available due to the use of the decoupled natural orthogonal complement (DeNOC) matrices for flexible robots that are proposed in this paper. The order of the proposed algorithm is $O(n)+O(m^3/3)$, where n is the total number of links and m is the number of modes in which each link is assumed to vibrate. The numerical stability aspects of the proposed forward dynamics algorithm for the flexible link robotic systems are studied here using the following two schemes: one based on power drift of the system, and other using the time duration for which the simulation results match with the desired results. The paper is organized as follows: In Sect. 2, kinematic description of a flexible link is given. Some definitions are introduced in Sect. 3, followed by the formulation of the dynamic model in Sect. 4. In Sect. 5, a recursive the forward dynamics algorithm for serial flexible robotic systems is proposed, whereas the simulation results of a flexible two link arm and a spatial Canadarm with two links flexible are reported in Sect. 6. The numerical stability of the proposed algorithm is then investigated in Sect. 7, followed by the conclusions in Sect. 8.

2 Kinematic description

Figure 1 shows a serial robotic system having a fixed base and n-moving bodies, which are either rigid or flexible. Figure 2 shows the ith flexible link. For simplicity, and without any loss of generality, each flexible link is assumed to vibrate in its m_ith mode in bending and $\bar{m}_i$ modes in torsion. Hence, the degree of freedom (DOF) of the system is,

$$\bar{n} \equiv n + \sum_{i=1}^{n_f} \left(3m_i + \bar{m}_i\right),$$

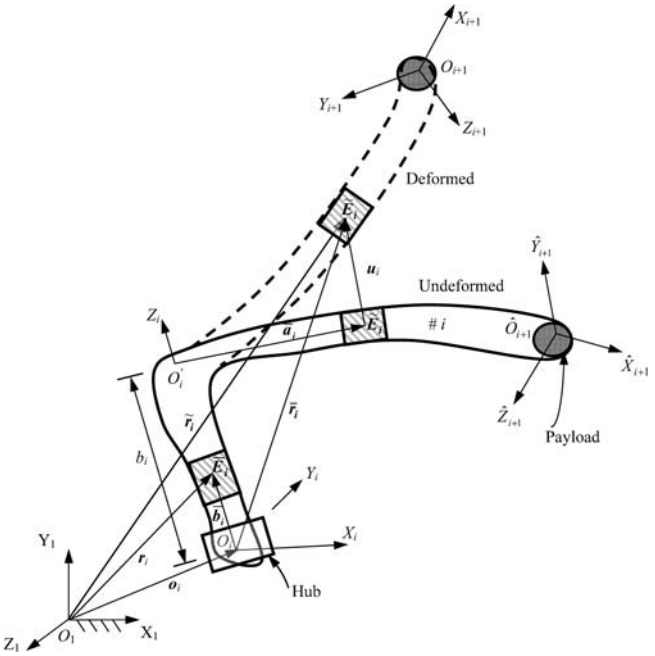


Fig. 2 The ith flexible link

where $n = n_r + n_f$, n_r and n_f are the number of rigid and flexible links, respectively. For the kinematic description of the elastic deformation of each flexible link, the AMM [30, 49] is used. Thus, the deformation of any element $\tilde{E}_i$ lying along $\hat{X}_{i+1}$ axis of the link, Fig. 2, due to bending is given by the 3-dimensional vector, $\boldsymbol{u}_i$, as

$$\boldsymbol{u}_i(\bar{\boldsymbol{a}}_i, t) \equiv \begin{bmatrix} u_i^x & u_i^y & u_i^z \end{bmatrix}^{\mathrm{T}}, \tag{1a}$$

where u_i^x, u_i^y, and u_i^z are the projections of the deflection vector $\boldsymbol{u}_i$ on the $\hat{X}_{i+1}$, $\hat{Y}_{i+1}$, and $\hat{Z}_{i+1}$ axes, respectively, and $\bar{\boldsymbol{a}}_i$ is the position vector of element $\tilde{E}_i$ from O_i'. Thus, $\bar{\boldsymbol{a}}_i = \bar{a}_i \hat{\boldsymbol{x}}_{i+1}$, where $\hat{\boldsymbol{x}}_{i+1}$ is the unit vector along $\hat{X}_{i+1}$ axis and $\bar{a}_i$ is the axial distance of $\tilde{E}_i$ from O_i' along $\hat{X}_{i+1}$ of the ith link. Note that $\bar{a}_i$ varies from 0 to a_i—one of the DH parameters of the link, as defined in Appendix A. Moreover, $(\bar{\boldsymbol{a}}_i, t)$ denotes the space and time dependence of the vector $\boldsymbol{u}_i$. The term u_i^x results in centrifugal stiffening of the link [8, 30, 44], which is significant in the analysis of flexible multibody systems, when the angular rates of the bodies are considerably greater than their first natural frequency. The centrifugal stiffening effect becomes significant only when the flexible link rotates at a very large angular rate [12]. Hence, for a flexible link used in industrial robots and in satellites whose speed of operation are generally slow, the effect of centrifugal stiffening is neglected. Moreover, flexibility along the joint axis, Z_i, is ignored due to the assumption of ith link rigid along Z_i, i.e., $O_i O_i'$ of Fig. 2. The above two assumptions are quite common in the literature [13, 14, 49]. For a prismatic joint, the flexibility along the Z_i-axis is also ignored. Otherwise, the smooth translation along the axis is not possible. Now, using AMM, vector, $\boldsymbol{u}_i$, can be

expressed in terms of space dependent eigen functions and time dependent amplitudes as

$$u_i^x = \begin{cases} 0 & \text{for revolute joints,} \\ \sum_{j=1}^{m_i} s_{i,j}^x d_{i,j}^x & \text{for prismatic joints,} \end{cases}$$

$$u_i^y = \sum_{j=1}^{m_i} s_{i,j}^y d_{i,j}^y \quad \text{for revolute and prismatic joints,} \tag{1b}$$

$$u_i^z = \begin{cases} \sum_{j=1}^{m_i} s_{i,j}^z d_{i,j}^z & \text{for revolute joints,} \\ 0 & \text{for prismatic joints.} \end{cases}$$

In order to express (1b) in matrix-vector form, the three $3 \times m_i$ matrices, namely, $\boldsymbol{S}_i^x$, $\boldsymbol{S}_i^y$, and $\boldsymbol{S}_i^z$, are introduced as follows:

$$\boldsymbol{S}_i^x \equiv \begin{bmatrix} s_{i,1}^x \dots s_{i,m_i}^x \\ 0 \dots 0 \\ 0 \dots 0 \end{bmatrix}; \quad \boldsymbol{S}_i^y \equiv \begin{bmatrix} 0 \dots 0 \\ s_{i,1}^y \dots s_{i,m_i}^y \\ 0 \dots 0 \end{bmatrix}; \quad \text{and} \quad \boldsymbol{S}_i^z \equiv \begin{bmatrix} 0 \dots 0 \\ 0 \dots 0 \\ s_{i,1}^z \dots s_{i,m_i}^z \end{bmatrix}, \tag{2a}$$

where $s_{i,1}^x \dots s_{i,m_i}^x$ denote the shape functions along the $\hat{X}_{i+1}$ axis corresponding to the m_i modes. Similarly, $s_{i,1}^y \dots s_{i,m_i}^y$ and $s_{i,1}^z \dots s_{i,m_i}^z$ represent the shape functions of the link along $\hat{Y}_{i+1}$ and $\hat{Z}_{i+1}$ axes in m_i modes, respectively. The overall shape function matrix of the ith flexible beam, vibrating in m_i modes is thus given by

$$\boldsymbol{S}_i \equiv \begin{bmatrix} \boldsymbol{S}_i^x & \boldsymbol{S}_i^y & \boldsymbol{S}_i^z \end{bmatrix}. \tag{2b}$$

Moreover, three m_i-dimensional vectors, $\boldsymbol{d}_i^x$, $\boldsymbol{d}_i^y$ and $\boldsymbol{d}_i^z$, are introduced as

$$\boldsymbol{d}_i^x \equiv \begin{bmatrix} d_{i,1}^x & \dots & d_{i,m_i}^x \end{bmatrix}^{\mathrm{T}}; \quad \boldsymbol{d}_i^y \equiv \begin{bmatrix} d_{i,1}^y & \dots & d_{i,m_i}^y \end{bmatrix}^{\mathrm{T}}; \quad \text{and} \quad \boldsymbol{d}_i^z \equiv \begin{bmatrix} d_{i,1}^z & \dots & d_{i,m_i}^z \end{bmatrix}^{\mathrm{T}}, \tag{2c}$$

where the vectors, $\boldsymbol{d}_i^x$, $\boldsymbol{d}_i^y$, and $\boldsymbol{d}_i^z$, are respectively the vectors of time dependent amplitudes corresponding to the shape function along $\hat{X}_{i+1}$, $\hat{Y}_{i+1}$, and $\hat{Z}_{i+1}$ axes, (2a). The components of $\boldsymbol{d}_i^x$, $\boldsymbol{d}_i^y$, and $\boldsymbol{d}_i^z$, are treated here as the generalized coordinates to describe the bending deflection of the ith link, along with those associated with the angular rotation of the ith joint, namely, θ_i. Next, the $3m_i$-dimensional vector of the time dependent amplitudes, $\boldsymbol{d}_i$, is defined by

$$\boldsymbol{d}_i \equiv \begin{bmatrix} \begin{bmatrix} \boldsymbol{d}_i^x \end{bmatrix}^{\mathrm{T}} & \begin{bmatrix} \boldsymbol{d}_i^y \end{bmatrix}^{\mathrm{T}} & \begin{bmatrix} \boldsymbol{d}_i^z \end{bmatrix}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}. \tag{2d}$$

Combining (2b–2d), (1b) is expressed as

$$\boldsymbol{u}_i = \boldsymbol{S}_i \boldsymbol{d}_i \tag{2e}$$

where the 3-dimensional vector, $\boldsymbol{u}_i$, is defined in (1a). Furthermore, since the flexible part of the link can also undergo torsion about $\hat{X}_{i+1}$ axis, it can be discretized similar to the bending deformation as

$$\beta_i \equiv \bar{\boldsymbol{s}}_i^{\mathrm{T}} \boldsymbol{c}_i \tag{3a}$$

where β_i is the scalar angular deformation of the cross-section of the element, $\tilde{E}_i$, and the $\bar{m}_i$-dimensional vectors, $\bar{\boldsymbol{s}}_i$ and $\boldsymbol{c}_i$, are the shape functions due to torsion and the time-dependent torsional amplitudes, respectively. Vectors $\bar{\boldsymbol{s}}_i$ and $\boldsymbol{c}_i$ are defined by

$$\bar{\boldsymbol{s}}_i \equiv \begin{bmatrix} \bar{s}_{i,1} & \dots & \bar{s}_{i,\bar{m}_i} \end{bmatrix}^{\mathrm{T}}; \qquad \boldsymbol{c}_i \equiv \begin{bmatrix} c_{i,1} & \dots & c_{i,\bar{m}_i} \end{bmatrix}^{\mathrm{T}}. \tag{3b}$$

Note that the components of $\boldsymbol{c}_i$ will also be treated as the generalized coordinates for the ith link, motion, along with $\boldsymbol{d}_i$ and θ_i.

3 Some definitions

Referring to the serial robot with flexible links under study, Figs. 1 and 2, the following definitions are introduced:

- $\boldsymbol{t}_i$ and $\boldsymbol{w}_i$: The $(6+3m_i+\bar{m}_i)$-dimensional twist and wrench of the ith flexible link, i.e.,

$$\boldsymbol{t}_i \equiv \begin{bmatrix} \boldsymbol{v}_i^{\mathrm{T}} & \boldsymbol{\omega}_i^{\mathrm{T}} & \dot{\boldsymbol{d}}_i^{\mathrm{T}} & \dot{\boldsymbol{c}}_i \end{bmatrix}^{\mathrm{T}}; \qquad \boldsymbol{w}_i \equiv \begin{bmatrix} \boldsymbol{f}_i^{\mathrm{T}} & \boldsymbol{n}_i^{\mathrm{T}} & \boldsymbol{\varepsilon}_i^{\mathrm{T}} & \boldsymbol{\vartheta}_i \end{bmatrix}^{\mathrm{T}}, \tag{4}$$

  where, $\boldsymbol{v}_i$ and $\boldsymbol{\omega}_i$, are the 3-dimensional vectors of velocity of the point, O_i, of the ith link, and its angular velocity, respectively. Moreover, the $3m_i$-dimensional vector, $\dot{\boldsymbol{d}}_i$, and the $\bar{m}_i$-dimensional vector, $\dot{\boldsymbol{c}}_i$, are the time derivatives of the time dependent variables $\boldsymbol{d}_i$ and $\boldsymbol{c}_i$, defined in (2c) and (3b), respectively. The vectors, $\boldsymbol{f}_i$ and $\boldsymbol{n}_i$, are the force at O_i and the moment about O_i of the ith link, respectively, whereas $\boldsymbol{\varepsilon}_i$ is the $3m_i$-dimensional generalized force vector associated with the generalized coordinates $\boldsymbol{d}_i$ due to bending, and $\boldsymbol{\vartheta}_i$ is the $\bar{m}_i$-dimensional generalized force vector associated with the generalized coordinates $\boldsymbol{c}_i$ due to torsion. For a rigid robot, vectors $\boldsymbol{d}_i$, $\boldsymbol{c}_i$, $\boldsymbol{\varepsilon}_i$ and $\boldsymbol{\vartheta}_i$ vanish and the dimension of the vectors, $\boldsymbol{t}_i$ and $\boldsymbol{w}_i$, reduce to six [1, 7].
- $\boldsymbol{t}$ and $\boldsymbol{w}$: The $\hat{n} \equiv 6n_r + (6+3m_i+\bar{m}_i)n_f$-dimensional vector of generalized twist and wrench, respectively, which are defined as

$$\boldsymbol{t} \equiv \begin{bmatrix} \boldsymbol{t}_1^{\mathrm{T}} & \boldsymbol{t}_2^{\mathrm{T}} & \dots & \boldsymbol{t}_n^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}; \qquad \boldsymbol{w} \equiv \begin{bmatrix} \boldsymbol{w}_1^{\mathrm{T}} & \boldsymbol{w}_2^{\mathrm{T}} & \dots & \boldsymbol{w}_n^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}, \tag{5}$$

  where, $\boldsymbol{t}_i$ and $\boldsymbol{w}_i$, for $i = 1, \dots, n$, are given by (4).
- $\boldsymbol{q}_i$ and $\boldsymbol{\tau}_i$: The $(1+3m_i+\bar{m}_i)$-dimensional vectors of joint-displacements and amplitudes (JDA), and the corresponding generalized forces of the ith flexible link, i.e.,

$$\boldsymbol{q}_i \equiv \begin{bmatrix} \theta_i & \boldsymbol{d}_i^{\mathrm{T}} & \boldsymbol{c}_i^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \quad \text{and} \quad \boldsymbol{\tau}_i \equiv \begin{bmatrix} \tau_i & \boldsymbol{\varepsilon}_i^{\mathrm{T}} & \boldsymbol{\vartheta}_i^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}, \tag{6}$$

  where θ_i is the rotational or translational displacement of the ith joint depending on its type, i.e., revolute or prismatic, respectively, and vectors $\boldsymbol{d}_i$ and $\boldsymbol{c}_i$ are defined in (2c) and (3b), respectively. Moreover, τ_i, and the vectors, $\boldsymbol{\varepsilon}_i$ and $\boldsymbol{\vartheta}_i$, are respectively the generalized forces corresponding to the joint coordinate, θ_i, and the amplitude vectors, $\boldsymbol{d}_i$ and $\boldsymbol{c}_i$. Note that for a rigid body $\boldsymbol{\varepsilon}_i$ and $\boldsymbol{\vartheta}_i$ vanish, and $\boldsymbol{\tau}_i$ reduces to a scalar [1, 7].
- $\dot{\boldsymbol{q}}$ and $\boldsymbol{\tau}$: The $\bar{n}$-dimensional vector of rates of JDA vector and the vector of corresponding generalized forces, i.e.,

$$\dot{\boldsymbol{q}} \equiv \begin{bmatrix} \dot{\boldsymbol{q}}_1 & \dot{\boldsymbol{q}}_2 & \dots & \dot{\boldsymbol{q}}_n \end{bmatrix}^{\mathrm{T}} \quad \text{and} \quad \boldsymbol{\tau} \equiv \begin{bmatrix} \boldsymbol{\tau}_1 & \boldsymbol{\tau}_2 & \dots & \boldsymbol{\tau}_n \end{bmatrix}^{\mathrm{T}}, \tag{7}$$

where $\dot{\boldsymbol{q}}_i$ is the time-rate of change of the joint-and-amplitude vector of the ith flexible link, and $\boldsymbol{\tau}_i$ is the corresponding generalized forces of the ith flexible link. For a rigid link, $\boldsymbol{q}_i \equiv \theta_i$.

4 Dynamic modeling

In this section, the derivation of the decoupled natural orthogonal complement (DeNOC) matrices associated with the velocity constraints of the serial-chain robotic system with flexible links is outlined.

4.1 The DeNOC matrices for flexible robots

The DeNOC matrices for a serial flexible robots are derived below:

(1) The twist of the ith flexible link is expressed in terms of the $(i-1)$st one as

$$\boldsymbol{t}_i = \boldsymbol{A}_{i,i-1}\boldsymbol{t}_{i-1} + \boldsymbol{P}_i\dot{\boldsymbol{q}}_i, \tag{8}$$

where $\boldsymbol{t}_i$ is the $(6+3m_i+\bar{m}_i)$-dimensional twist vector of the ith flexible link defined in (4), and $\dot{\boldsymbol{q}}_i$ is the time-rate of change of vector $\boldsymbol{q}_i$ defined in (6), whereas the $(6+3m_i+\bar{m}_i)\times(6+3m_i+\bar{m}_i)$ twist propagation matrix, $\boldsymbol{A}_{i,i-1}$, and the $(6+3m_i+\bar{m}_i)\times(1+3m_i+\bar{m}_i)$ JDA propagation matrix, $\boldsymbol{P}_i$, for the flexible links are defined as

$$\boldsymbol{A}_{i,i-1} \equiv \begin{bmatrix} \boldsymbol{R}_{i,i-1} & \boldsymbol{F}_{i-1} \\ \tilde{\boldsymbol{O}} & \bar{\boldsymbol{O}} \end{bmatrix}; \qquad \boldsymbol{P}_i \equiv \begin{bmatrix} \boldsymbol{p}_i & \tilde{\boldsymbol{O}}^{\mathrm{T}} \\ \tilde{\boldsymbol{0}} & \bar{\boldsymbol{1}} \end{bmatrix}. \tag{9}$$

In (9), the 6×6 matrix, $\boldsymbol{R}_{i,i-1}$, and the 6-dimensional vector, $\boldsymbol{p}_i$, are defined as

$$\boldsymbol{R}_{i,i-1} \equiv \begin{bmatrix} \boldsymbol{1} & \boldsymbol{a}_{i,i-1}\times\boldsymbol{1} \\ \boldsymbol{O} & \boldsymbol{1} \end{bmatrix}; \quad \boldsymbol{p}_i \equiv \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{z}_i \end{bmatrix} \text{ for revolute}; \quad \boldsymbol{p}_i \equiv \begin{bmatrix} \boldsymbol{z}_i \\ \boldsymbol{0} \end{bmatrix} \text{ for prismatic} \tag{10}$$

in which, $\boldsymbol{a}_{i,i-1} \equiv -\boldsymbol{a}_{i-1,i}$, is the position vector of the point, O_i of the ith link, from O_{i-1} of the $(i-1)$st link. Moreover, the 3×3 cross product tensor, $\boldsymbol{a}_{i,i-1}\times\boldsymbol{1}$, associated with the vector, $\boldsymbol{a}_{i,i-1}$ [35] is defined such that $(\boldsymbol{a}_{i,i-1}\times\boldsymbol{1})\boldsymbol{x} = \boldsymbol{a}_{i,i-1}\times\boldsymbol{x}$, for any 3-dimensional Cartesian vector $\boldsymbol{x}$. Furthermore, $\boldsymbol{z}_i$ is the 3-dimensional unit vector along the axis of rotation of a revolute joint or along the direction of a prismatic joint, whereas $\boldsymbol{O}$ and $\boldsymbol{0}$ are respectively the 3×3 zero matrix, and the 3-dimensional vector of zeros, and $\boldsymbol{1}$ is the 3×3 identity matrix. Furthermore, the $6\times(3m_i+\bar{m}_i)$ matrix, $\boldsymbol{F}_{i-1}$, is given by

$$\boldsymbol{F}_{i-1} \equiv \begin{bmatrix} \boldsymbol{S}_{i-1} & \hat{\boldsymbol{O}}_{i-1} \\ \boldsymbol{\Delta}_{i-1} & \boldsymbol{C}_{i-1} \end{bmatrix} \tag{11a}$$

in which the $3\times 3m_i$ matrix, $\boldsymbol{S}_{i-1}$, contains the shape functions corresponding to the three dimensional bending deflections, as defined in (2a–b), whereas the $3\times 3m_i$ matrix, $\boldsymbol{\Delta}_{i-1}$, contains the first derivatives of the bending shape functions corresponding to the $(i-1)$st flexible link. For the ith flexible link, if s_{ij} denotes the shape function corresponding to the

jth mode, its first derivative with respect to its length of the flexible part, $\bar{a}_i$, evaluated at the link end, a_i, is given by $\partial s_{ij}/\partial \bar{a}_i|_{a_i}$. Accordingly, the matrix $\boldsymbol{\Delta}_i$ is represented as

$$\boldsymbol{\Delta}_i \equiv \begin{bmatrix} 0\ldots0 & 0\ldots0 & 0\ldots0 \\ 0\ldots0 & 0\ldots0 & -\dfrac{\partial s^y_{i,1}}{\partial \bar{a}_i}\Big|_{a_i} \ldots -\dfrac{\partial s^y_{i,m_i}}{\partial \bar{a}_i}\Big|_{a_i} \\ 0\ldots0 & \dfrac{\partial s^z_{i,1}}{\partial \bar{a}_i}\Big|_{a_i} \ldots \dfrac{\partial s^z_{i,m_i}}{\partial \bar{a}_i}\Big|_{a_i} & 0\ldots0 \end{bmatrix} \tag{11b}$$

in which $s^y_{i,j}$ and $s^z_{i,j}$, for $j = 1, \ldots, m_i$ are those appeared in (11b). Moreover, the $3 \times \bar{m}_i$ matrix $\boldsymbol{C}_{i-1}$, contains the shape functions corresponding to the torsion of $(i-1)$st flexible link in $\bar{m}_i$ modes. Accordingly, the matrix $\boldsymbol{C}_i$ is represented as

$$\boldsymbol{C}_i \equiv \begin{bmatrix} \bar{s}_{i,1}|_{a_i} \ldots \bar{s}_{i,\bar{m}_i}|_{a_i} \\ 0\ldots0 \\ 0\ldots0 \end{bmatrix}, \tag{11c}$$

where $\bar{s}_{i,k}$ for $k = 1, \ldots, \bar{m}_i$, are the shape functions of the beam in torsional vibration, as defined in (3b). Note that $\tilde{\boldsymbol{O}}$, $\bar{\boldsymbol{O}}$, and $\hat{\boldsymbol{O}}$ in (9–11) are respectively the $(3m_i + \bar{m}_i) \times 6$, $(3m_i + \bar{m}_i) \times (3m_i + \bar{m}_i)$, and $3 \times \bar{m}_i$ zero matrices. Also, $\tilde{\mathbf{0}}$ and $\bar{\mathbf{1}}$ are respectively the $(3m_i + \bar{m}_i)$-dimensional vectors of zeros and the $(3m_i + \bar{m}_i) \times (3m_i + \bar{m}_i)$ identity matrix. In line with the rigid link system [35], the $(6+3m_i+\bar{m}_i) \times (6+3m_i+\bar{m}_i)$ matrix, $\boldsymbol{A}_{i,i-1}$, is termed here as the twist propagation matrix for the flexible link which satisfies the property, $\boldsymbol{A}_{i,j}\boldsymbol{A}_{j,k} = \boldsymbol{A}_{i,k}$. Similarly, the $(6 + 3m_i + \bar{m}_i) \times (1 + 3m_i + \bar{m}_i)$ matrix, $\boldsymbol{P}_i$, is termed as the JDA propagation matrix.

(2) Now, for the n-link serial flexible link robot, Figs. 1 and 2, the $\hat{n}$-dimensional vector of generalized twist, $\boldsymbol{t}$ of (5) can be expressed using (8) as

$$\boldsymbol{t} = \boldsymbol{A}\boldsymbol{t} + \boldsymbol{N}_d\dot{\boldsymbol{q}}, \tag{12}$$

where $\boldsymbol{q}$ is the $\bar{n}$-dimensional vector of the JDA of the robot. In (12), the $\hat{n} \times \hat{n}$ matrix, $\boldsymbol{A}$, and the $\hat{n} \times \bar{n}$ matrix, $\boldsymbol{N}_d$, are given by

$$\boldsymbol{A} \equiv \begin{bmatrix} \boldsymbol{O} & \ldots & \ldots & \boldsymbol{O} \\ \boldsymbol{A}_{21} & \boldsymbol{O} & \ldots & \boldsymbol{O} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O} & \ldots & \boldsymbol{A}_{n,n-1} & \boldsymbol{O} \end{bmatrix}; \qquad \boldsymbol{N}_d \equiv \begin{bmatrix} \boldsymbol{P}_1 & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{P}_2 & \ldots & \mathbf{0} \\ \vdots & \vdots & \ddots & \\ \mathbf{0} & \mathbf{0} & \ldots & \boldsymbol{P}_n \end{bmatrix}, \tag{13}$$

where $\boldsymbol{O}$ and $\mathbf{0}$ are the $(6 + 3m_i + \bar{m}_i) \times (6 + 3m_i + \bar{m}_i)$ matrix and the $(6 + 3m_i + \bar{m}_i)$-dimensional vector of zeros, respectively. Henceforth, $\boldsymbol{O}$ and $\mathbf{0}$ should be understood as of compatible dimensions based on the expressions where they appear.

(3) Equation (12) is rearranged and written as

$$\boldsymbol{t} = \boldsymbol{N}\dot{\boldsymbol{\theta}}, \quad \text{where } \boldsymbol{N} \equiv \boldsymbol{N}_l\boldsymbol{N}_d. \tag{14}$$

In (14), the $\hat{n} \times \hat{n}$ matrix, $\boldsymbol{N}_l$ is given by:

$$\boldsymbol{N}_l \equiv \begin{bmatrix} \mathbf{1} & \boldsymbol{O} & \dots & \boldsymbol{O} \\ \boldsymbol{A}_{21} & \mathbf{1} & \dots & \boldsymbol{O} \\ \vdots & \vdots & \ddots & \\ \boldsymbol{A}_{n1} & \boldsymbol{A}_{n2} & \dots & \mathbf{1} \end{bmatrix}, \tag{15}$$

where $\mathbf{1}$ denotes the $(6+3m_i+\bar{m}_i) \times (6+3m_i+\bar{m}_i)$ identity matrix. Like $\boldsymbol{O}$ and $\mathbf{0}$, henceforth, $\mathbf{1}$ should be understood as of compatible size based on where it appears. The matrix, $\boldsymbol{N}$, in a coupled form is the natural orthogonal complement (NOC) matrix for the serial flexible robot as reported in [13]. In this paper, the decoupled form, namely, $\boldsymbol{N}_l$ and $\boldsymbol{N}_d$ matrices, are derived for the flexible systems for the first time, which are referred as the decoupled natural orthogonal complement (DeNOC) matrices for flexible robots. The DeNOC matrices allow one to write the matrix and vector elements associated with the dynamic equations of motion in analytical form leading to recursive forward dynamics algorithm.

4.2 Dynamic modeling of flexible robots

The dynamic modeling of the flexible robot, shown in Fig. 1, is now derived using the equivalence of EL and NE methodology, as proposed for rigid robots in [7], and the DeNOC matrices for the flexible link robots derived in Sect. 4.1. The steps are outlined below:

(1) Referring to Fig. 2, the position vectors of the elements, $\bar{E}_i$, $\tilde{E}_i$, and payload of mass m_{pi} on the ith link, namely, $\boldsymbol{r}_i$, $\tilde{\boldsymbol{r}}_i$, and $\boldsymbol{r}_{pi}$ are respectively given by

$$\begin{aligned} \boldsymbol{r}_i &= \boldsymbol{o}_i + \bar{\boldsymbol{b}}_i, \quad \text{where } \bar{\boldsymbol{b}}_i = \bar{b}_i \boldsymbol{z}_i, \\ \tilde{\boldsymbol{r}}_i &= \boldsymbol{o}_i + \bar{\boldsymbol{r}}_i, \quad \text{where } \bar{\boldsymbol{r}}_i = b_i \boldsymbol{z}_i + \bar{a}_i \hat{\boldsymbol{x}}_{i+1} + \boldsymbol{u}_i, \text{ and} \\ \boldsymbol{r}_{pi} &= \boldsymbol{o}_i + \bar{\boldsymbol{r}}_{pi}, \quad \text{where } \bar{\boldsymbol{r}}_{pi} = b_i \boldsymbol{z}_i + a_i \hat{\boldsymbol{x}}_{i+1} + \boldsymbol{u}_{pi}, \end{aligned} \tag{16}$$

where, a_i and b_i are the DH-parameters of the link, as defined in Appendix A, $\bar{b}_i$ is the axial distance of element $\bar{E}_i$ along Z_i from O_i, and $\bar{a}_i$ is the axial distance of element $\tilde{E}_i$ along $\hat{X}_{i+1}$ from O_i', as shown in Fig. 2. Note that $\bar{\boldsymbol{b}}_i$ is the position vector of element $\bar{E}_i$ along Z_i from O_i, whose magnitude is $\bar{b}_i$. The term, $\bar{b}_i$, varies from 0 to b_i, and $\bar{a}_i$ varies from 0 to a_i. Moreover, the unit vectors along Z_i and $\hat{X}_{i+1}$-axes are denoted with $\boldsymbol{z}_i$ and $\hat{\boldsymbol{x}}_{i+1}$, respectively, and vector $\boldsymbol{o}_i$ denotes the position vector of the point, O_i, of the ith frame with respect to the origin of the fixed first frame. Furthermore, vectors $\boldsymbol{u}_i$ and $\boldsymbol{u}_{pi}$ are respectively the positions of the element, $\tilde{E}_i$, and the payload m_{pi}, on the deformed flexible link from its undeformed state. Vector $\boldsymbol{u}_i$ is indicated in Fig. 2. Note that the payload is considered as a concentrated point mass at the tip of the link that accounts for any assembly with sensors attached to the ith link. For the nth link, it is the real load to be carried by it.

(2) The kinetic energy, T_i, for the ith flexible link is then given by

$$T_i = \frac{1}{2}\int_0^{b_i} \rho_i \dot{\boldsymbol{r}}_i^{\mathrm{T}} \dot{\boldsymbol{r}}_i \, \mathrm{d}\bar{b}_i + \frac{1}{2}\int_0^{a_i} \rho_i \dot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}} \dot{\tilde{\boldsymbol{r}}}_i \, \mathrm{d}\bar{a}_i + \frac{1}{2} m_{pi} \dot{\boldsymbol{r}}_{pi}^{\mathrm{T}} \dot{\boldsymbol{r}}_{pi} + \frac{1}{2}\int_0^{a_i} \rho_i I_{pi} \dot{\beta}_i^2 \, \mathrm{d}x_{i+1} + T_{hi}, \tag{17}$$

where ρ_i is the mass per unit length of the ith link, and the vectors, $\dot{\boldsymbol{r}}_i$, $\dot{\tilde{\boldsymbol{r}}}_i$, and $\dot{\boldsymbol{r}}_{pi}$, are the velocities of the elements, $\bar{E}_i$, $\tilde{E}_i$, and payload, respectively, which can be written from (16)

as

$$\begin{aligned}\dot{\boldsymbol{r}}_i &= \boldsymbol{v}_i + \boldsymbol{\omega}_i \times \bar{\boldsymbol{b}}_i;\\ \dot{\tilde{\boldsymbol{r}}}_i &= \boldsymbol{v}_i + \boldsymbol{\omega}_i \times \bar{\boldsymbol{r}}_i + \dot{b}_i \boldsymbol{z}_i + \dot{\boldsymbol{u}}_i; \quad \text{and}\\ \dot{\boldsymbol{r}}_{pi} &= \boldsymbol{v}_i + \boldsymbol{\omega}_i \times \bar{\boldsymbol{r}}_{pi} + \dot{b}_i \boldsymbol{z}_i + \dot{\boldsymbol{u}}_{pi},\end{aligned} \tag{18}$$

where, $\boldsymbol{v}_i$ is substituted for $\dot{\boldsymbol{o}}_i$, i.e., $\boldsymbol{v}_i \equiv \dot{\boldsymbol{o}}_i$. Moreover, $\dot{\bar{a}}_i = \dot{a}_i = 0$ is used in (18) due to the assumption of no extension along $\hat{X}_{i+1}$ axis of the ith flexible link shown in Fig. 2. Similarly, $\dot{\bar{b}}_i = 0$, as this portion of the link is assumed rigid. Furthermore, $\dot{b}_i$ represents the linear joint rate in the presence of prismatic joint. However, for the revolute joint, it vanishes, i.e., $\dot{b}_i = 0$. Also, the scalar, I_{pi}, denotes the polar moment of inertia of the cross-section of the element, $\tilde{E}_i$, belonging to the ith flexible link, whereas β_i is the angular deformation of the cross-section of the element, $\tilde{E}_i$, as defined after (3a). Finally, the term, T_{hi}, represents the kinetic energy due to the hub inertia at the joint, which is given by,

$$T_{hi} = \frac{1}{2}\boldsymbol{\omega}_i^{\mathrm{T}} \boldsymbol{I}_{hi} \boldsymbol{\omega}_i, \tag{19}$$

where $\boldsymbol{I}_{hi}$ is the 3×3 inertia tensor for the hub. A hub includes the effect of motor and the gear assembly located at the joints.

(3) The EL equations of motion for the whole system are then given by [30]:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{\boldsymbol{q}}_i}\right) - \frac{\partial T}{\partial \boldsymbol{q}_i} = \boldsymbol{\tau}_i, \quad \text{for } i = 1, \ldots, n, \tag{20}$$

where n is the total number of bodies, whereas $\boldsymbol{q}_i$ is the $(1 + 3m_i + \bar{m}_i)$-dimensional vector of independent generalized coordinates defined in (6). Accordingly, vector $\boldsymbol{\tau}_i$ is the associated generalized forces given by $\boldsymbol{\tau}_i \equiv \boldsymbol{\tau}_i^E + \boldsymbol{\tau}_i^s$, in which $\boldsymbol{\tau}_i^E$ is the generalized forces due to external forces and moments on the whole system, and $\boldsymbol{\tau}_i^s$ is the generalized forces due to strains in the ith link. Vector $\boldsymbol{\tau}_i^s$ has the following form:

$$\boldsymbol{\tau}_i^s \equiv \partial V_s / \partial \boldsymbol{q}_i, \tag{20a}$$

where V_s is the potential energy of the system at hand due to strains. Note here that the effect of potential energy due to gravity is included by adding negative of the acceleration due to gravity to the linear acceleration of the first link, $\dot{\boldsymbol{v}}_1$, as proposed in [52]. Potential energy of the system at hand due to strain energy, V_s as required in (20a) is evaluated as:

$$\begin{aligned}V_s &= \sum_{i=1}^{n_f} V_{si}, \quad \text{where}\\ V_{si} &\equiv \frac{1}{2}\int_0^{a_i} E_i I_i^y \left(\frac{\partial^2 u_i^y}{\partial \bar{a}_i^2}\right)^2 \mathrm{d}\bar{a}_i + \frac{1}{2}\int_0^{a_i} E_i I_i^z \left(\frac{\partial^2 u_i^z}{\partial \bar{a}_i^2}\right)^2 \mathrm{d}\bar{a}_i\\ &\quad + \frac{1}{2}\int_0^{a_i} G_i I_{pi} \left(\frac{\partial^2 \beta_i}{\partial \bar{a}_i^2}\right)^2 \mathrm{d}\bar{a}_i,\end{aligned} \tag{20b}$$

where $E_i I_i^y$, $E_i I_i^z$ are the flexure rigidity, and $G_i I_{pi}$ is the torsional rigidity of the ith link in which E_i is the Young's modulus of elasticity, I_i^y and I_i^z are the moment of inertia of the ith link about its $\hat{Y}_{i+1}$ and $\hat{Z}_{i+1}$ axes, respectively, and I_{pi} is the polar moment of inertia of the cross-section of the link. The deflection of the link, u_i^y, u_i^z, β_i, are given by (3a), respectively,

whereas the deflection along X_{i+1} axis is ignored. Now, the partial differentiation of V_s with respect to $\boldsymbol{q}_i$, as required in (20a) is evaluated. Note that the vector of generalized coordinates, $\boldsymbol{q}_i$ is the array of $\theta_i, \boldsymbol{d}_i^y, \boldsymbol{d}_i^z$, and $\boldsymbol{c}_i$, but V_s is a function of only $\boldsymbol{d}_i^y, \boldsymbol{d}_i^z$, and $\boldsymbol{c}_i$. As a result, $\boldsymbol{\tau}_i^s \equiv \partial V_s / \partial \boldsymbol{q}_i$ is obtained as

$$\boldsymbol{\tau}_i^s = \begin{bmatrix} 0 & \mathbf{0}^{\mathrm{T}} & \left[\boldsymbol{\tau}_i^{sy}\right]^{\mathrm{T}} & \left[\boldsymbol{\tau}_i^{sz}\right]^{\mathrm{T}} & \left[\boldsymbol{\tau}_i^{sx}\right]^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}. \tag{20c}$$

Since strain energy due to axial centrifugal stiffening is neglected, the m_i-dimensional zero vector $\mathbf{0}$ appears in $\boldsymbol{\tau}_i^s$ after the scalar zero of (20c). In (20c), the m_i-dimensional vectors $\boldsymbol{\tau}_i^{sy}$ and $\boldsymbol{\tau}_i^{sz}$, and the $\bar{m}_i$-dimensional vector, $\boldsymbol{\tau}_i^{sx}$, are given as

$$\begin{aligned} \boldsymbol{\tau}_i^{sy} &\equiv \begin{bmatrix} \tau_{i1}^{sy} & \dots & \tau_{im_i}^{sy} \end{bmatrix}^{\mathrm{T}}; \quad \boldsymbol{\tau}_i^{sz} \equiv \begin{bmatrix} \tau_{i1}^{sz} & \dots & \tau_{im_i}^{sz} \end{bmatrix}^{\mathrm{T}} \quad \text{and} \\ \boldsymbol{\tau}_i^{sx} &\equiv \begin{bmatrix} \tau_{i1}^{sx} & \dots & \tau_{i\bar{m}_i}^{sx} \end{bmatrix}^{\mathrm{T}}, \end{aligned} \tag{20d}$$

where τ_{ij}^{sy} and τ_{ij}^{sz} are defined, for $j = 1, \dots, m_i$, as

$$\begin{aligned} \tau_{ij}^{sy} &\equiv E_i I_i^y \int_0^{a_i} \left(\tilde{k}_{ij}^y \sum_{h=1}^{m_i} \tilde{k}_{ih}^y d_{ih}^y \right) \mathrm{d}\bar{a}_i \quad \text{and} \\ \tau_{ij}^{sz} &\equiv E_i I_i^z \int_0^{a_i} \left(\tilde{k}_{ij}^z \sum_{h=1}^{m_i} \tilde{k}_{ih}^z d_{ih}^z \right) \mathrm{d}\bar{a}_i. \end{aligned} \tag{20e}$$

Similarly, τ_{il}^{sx} are given as, for $l = 1, \dots, \bar{m}_i$, as

$$\tau_{il}^{sx} = G_i I_{pi}^x \int_0^{a_i} \left(\bar{k}_{il}^x \sum_{h=1}^{\bar{m}_i} \bar{k}_{ih}^x c_{ih}^x \right) \mathrm{d}\bar{a}_i, \tag{20f}$$

where for $j = 1, \dots, m_i$, $\tilde{k}_{ij}^y \equiv \frac{\partial^2 s_{ij}^y}{\partial \bar{a}_i^2}$, $\tilde{k}_{ij}^z \equiv \frac{\partial^2 s_{ij}^z}{\partial \bar{a}_i^2}$, and for $l = 1, \dots, \bar{m}_i$, $\bar{k}_{il}^x \equiv \frac{\partial^2 \bar{s}_{il}^x}{\partial \bar{a}_i^2}$, in which s_{ij}^y and s_{ij}^z are the shape functions of the ith link in its jth mode of vibration about $\hat{Y}_{i+1}$ and $\hat{Z}_{i+1}$ axis, respectively, as defined after (2a), whereas $\bar{s}_{il}^x$ is the shape function of the ith link in its lth mode of torsion, as given by (3b). Note that $d_{i,h}^y$, $d_{i,h}^z$, and $c_{i,l}^x$ are the generalized coordinates of the system. The total kinetic energy of the system is then, $T = \sum_{i=1}^{n} T_i$—n being the total number of rigid and flexible links, n_r and n_f, respectively, i.e., $n \equiv n_r + n_f$. The dynamic equations of motion for the flexible robot, as derived in Appendix B are expressed as

$$\begin{bmatrix} \left(\dfrac{\partial \boldsymbol{t}_1}{\partial \dot{\boldsymbol{q}}_i}\right)^{\mathrm{T}} & \dots & \left(\dfrac{\partial \boldsymbol{t}_n}{\partial \dot{\boldsymbol{q}}_i}\right)^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} w_1^* \\ \vdots \\ w_n^* \end{bmatrix} = \boldsymbol{\tau}_i, \quad \text{for } i = 1, \dots, n, \tag{21}$$

where the $(6+3m_i+\bar{m}_i)\times(1+3m_i+\bar{m}_i)$ matrix, $\partial \boldsymbol{t}_i/\partial\dot{\boldsymbol{q}}_j$, and the $(6+3m_i+\bar{m}_i)$-dimensional vector, $\boldsymbol{w}_i^*$, for $i=1,\ldots,n$, are reproduced from (B.10) as

$$\frac{\partial \boldsymbol{t}_i}{\partial \dot{\boldsymbol{q}}_j} \equiv \begin{bmatrix} \dfrac{\partial \boldsymbol{v}_i}{\partial \dot{\boldsymbol{q}}_j} \\ \dfrac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j} \\ \dfrac{\partial \dot{\boldsymbol{d}}_i}{\partial \dot{\boldsymbol{q}}_j} \\ \dfrac{\partial \dot{\boldsymbol{c}}_i}{\partial \dot{\boldsymbol{q}}_j} \end{bmatrix};$$

$$\boldsymbol{w}_i^* \equiv \begin{bmatrix} \displaystyle\int_0^{b_i} \rho_i \ddot{\boldsymbol{r}}_i \,\mathrm{d}\bar{b}_i + \int_0^{a_i} \rho_i \ddot{\bar{\boldsymbol{r}}}_i \,\mathrm{d}\bar{a}_i + m_{pi}\ddot{\boldsymbol{r}}_{pi} \\ \displaystyle\int_0^{b_i} \rho_i \bar{b}_i (\boldsymbol{z}_i \times \ddot{\boldsymbol{r}}_i)\,\mathrm{d}\bar{b}_i + \int_0^{a_i} \rho_i (\bar{\boldsymbol{r}}_i \times \ddot{\bar{\boldsymbol{r}}}_i)\,\mathrm{d}\bar{a}_i + m_{pi}(\bar{\boldsymbol{r}}_{pi} \times \ddot{\boldsymbol{r}}_{pi}) + (\boldsymbol{I}_{hi}\dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times \boldsymbol{I}_{hi}\boldsymbol{\omega}_i) \\ \displaystyle\int_0^{a_i} \rho_i \boldsymbol{S}_i^{\mathrm{T}} \ddot{\bar{\boldsymbol{r}}}_i \,\mathrm{d}\bar{a}_i + m_{pi}\boldsymbol{S}_i|_{a_i}^{\mathrm{T}} \ddot{\boldsymbol{r}}_{pi} \\ \displaystyle\int_0^{a_i} \rho_i I_{pi} \bar{\boldsymbol{s}}_i \ddot{\bar{\boldsymbol{c}}}_i^{\mathrm{T}} \bar{\boldsymbol{s}}_i \,\mathrm{d}\bar{a}_i \end{bmatrix}, \tag{22}$$

where $\boldsymbol{S}_i|_{a_i}$ implies the value of the shape function $\boldsymbol{S}_i$, (2b) evaluated at $\bar{a}_i = a_i$.

(4) The vector, $\boldsymbol{w}_i^*$ of (22), can be physically interpreted as the inertia wrench of the flexible link, and can be written as

$$\boldsymbol{w}_i^* = \boldsymbol{M}_i \dot{\boldsymbol{t}}_i + \boldsymbol{\gamma}_i, \tag{23}$$

where $\boldsymbol{M}_i$ is the $(6+3m_i+\bar{m}_i)\times(6+3m_i+\bar{m}_i)$ mass matrix, and $\boldsymbol{\gamma}_i$ is the $(6+3m_i+\bar{m}_i)$-dimensional vector. The mass matrix, $\boldsymbol{M}_i$, and vector $\boldsymbol{\gamma}_i$ are obtained as:

$$M_i \equiv \int_0^{b_i} \rho_i \begin{bmatrix} \mathbf{1} & -\bar{\boldsymbol{b}}_i \times \mathbf{1} & \boldsymbol{O} & \mathbf{0} \\ & -\bar{\boldsymbol{b}}_i \times (\bar{\boldsymbol{b}}_i \times \mathbf{1}) & \boldsymbol{O} & \mathbf{0} \\ & & \boldsymbol{O} & \mathbf{0} \\ \text{sym} & & & \mathbf{0} \end{bmatrix} \mathrm{d}\bar{b}_i$$

$$+ \int_0^{a_i} \rho_i \begin{bmatrix} \mathbf{1} & -\bar{\boldsymbol{r}}_i \times \mathbf{1} & \boldsymbol{S}_i & \mathbf{0} \\ & -\bar{\boldsymbol{r}}_i \times (\bar{\boldsymbol{r}}_i \times \mathbf{1}) & \bar{\boldsymbol{r}}_i \times \boldsymbol{S}_i & \mathbf{0} \\ & & \boldsymbol{S}_i^{\mathrm{T}} \boldsymbol{S}_i & \mathbf{0} \\ \text{sym} & & & I_{pi}\bar{\boldsymbol{s}}_i \bar{\boldsymbol{s}}_i^{\mathrm{T}} \end{bmatrix} \mathrm{d}\bar{a}_i$$

$$+ m_{pi}\begin{bmatrix} \mathbf{1} & -\bar{\boldsymbol{r}}_{pi}\times\mathbf{1} & \boldsymbol{S}_i|_{a_i} & \mathbf{0} \\ & -\bar{\boldsymbol{r}}_{pi}\times(\bar{\boldsymbol{r}}_{pi}\times\mathbf{1}) & \bar{\boldsymbol{r}}_{pi}\times\boldsymbol{S}_i|_{a_i} & \mathbf{0} \\ & & \boldsymbol{S}_i|_{a_i}^{\mathrm{T}}\boldsymbol{S}_i|_{a_i} & \mathbf{0} \\ \text{sym} & & & \mathbf{0} \end{bmatrix} + \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \mathbf{0} \\ & \boldsymbol{I}_{hi} & \boldsymbol{O} & \mathbf{0} \\ & & \boldsymbol{O} & \mathbf{0} \\ \text{sym} & & & \mathbf{0} \end{bmatrix},$$

$$\boldsymbol{\gamma}_i \equiv \int_0^{b_i} \rho_i \begin{bmatrix} \bar{\boldsymbol{\varpi}}_i \\ \bar{\boldsymbol{b}}_i\times\bar{\boldsymbol{\varpi}}_i \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \mathrm{d}\bar{b}_i + \int_0^{a_i} \rho_i \begin{bmatrix} \boldsymbol{\varpi}_i \\ \bar{\boldsymbol{r}}_i\times\boldsymbol{\varpi}_i \\ \boldsymbol{S}_i^{\mathrm{T}}\boldsymbol{\varpi}_i \\ \mathbf{0} \end{bmatrix} \mathrm{d}\bar{a}_i + m_{pi}\begin{bmatrix} \boldsymbol{\varpi}_{pi} \\ \bar{\boldsymbol{r}}_{pi}\times\boldsymbol{\varpi}_{pi} \\ \boldsymbol{S}_i|_{a_i}^{\mathrm{T}}\boldsymbol{\varpi}_{pi} \\ \mathbf{0} \end{bmatrix}$$

$$+ \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\omega}_i\times\boldsymbol{I}_{hi}\boldsymbol{\omega}_i \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \tag{24}$$

where $\bar{\boldsymbol{\varpi}}_i \equiv \boldsymbol{\omega}_i\times(\boldsymbol{\omega}_i\times\bar{\boldsymbol{b}}_i)$, $\boldsymbol{\varpi}_i \equiv \boldsymbol{\omega}_i\times[(\boldsymbol{\omega}_i\times\bar{\boldsymbol{r}}_i)+\dot{\boldsymbol{u}}_i]$, $\boldsymbol{\varpi}_{pi} \equiv \boldsymbol{\omega}_i\times[(\boldsymbol{\omega}_i\times\bar{\boldsymbol{r}}_{pi})+\dot{\boldsymbol{u}}_{pi}]$ and the word "sym" denotes the symmetric elements of the matrix, $\boldsymbol{M}_i$.

(5) Combining (23) for all n-links, i.e., $i=1,\ldots,n$, (21) can be written as

$$\left(\frac{\partial \boldsymbol{t}}{\partial \dot{\boldsymbol{q}}}\right)^{\mathrm{T}}\left(\boldsymbol{M}\dot{\boldsymbol{t}}+\boldsymbol{\gamma}\right)=\boldsymbol{\tau}, \tag{25}$$

where the $\hat{n}\times\hat{n}$ matrix, $\boldsymbol{M}$, the $\hat{n}$-dimensional vector, $\boldsymbol{\gamma}$, and the $\bar{n}$-dimensional vector, $\boldsymbol{\tau}$, are given by

$$M \equiv \mathrm{diag}\begin{bmatrix}\boldsymbol{M}_1 & \ldots & \boldsymbol{M}_n\end{bmatrix};\quad \boldsymbol{\gamma} \equiv \begin{bmatrix}\boldsymbol{\gamma}_1^{\mathrm{T}} & \ldots & \boldsymbol{\gamma}_n^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}};\quad \text{and}\quad \boldsymbol{\tau} \equiv \begin{bmatrix}\boldsymbol{\tau}_1 & \ldots & \boldsymbol{\tau}_n\end{bmatrix}^{\mathrm{T}}. \tag{26}$$

The vectors, $\boldsymbol{t}$ and $\boldsymbol{w}$, are defined in (5), and the $\hat{n}\times\hat{n}$ matrix, $\frac{\partial \boldsymbol{t}}{\partial \dot{\boldsymbol{q}}}$, is defined as

$$\frac{\partial \boldsymbol{t}}{\partial \dot{\boldsymbol{q}}} \equiv \begin{bmatrix} \frac{\partial \boldsymbol{t}_1}{\partial \dot{\boldsymbol{q}}_1} & \cdots & \frac{\partial \boldsymbol{t}_1}{\partial \dot{\boldsymbol{q}}_n} \\ \vdots & & \vdots \\ \frac{\partial \boldsymbol{t}_n}{\partial \dot{\boldsymbol{q}}_1} & \cdots & \frac{\partial \boldsymbol{t}_n}{\partial \dot{\boldsymbol{q}}_n} \end{bmatrix}. \tag{27}$$

From (14), it is clear that

$$\frac{\partial \boldsymbol{t}}{\partial \dot{\boldsymbol{q}}} = \boldsymbol{N}_l\boldsymbol{N}_d. \tag{28}$$

Hence, (25) yields

$$\boldsymbol{N}_d^{\mathrm{T}}\boldsymbol{N}_l^{\mathrm{T}}\left(\boldsymbol{M}\dot{\boldsymbol{t}}+\boldsymbol{\gamma}\right)=\boldsymbol{\tau}. \tag{29}$$

(6) Now, differentiating (5) with respect to time, one obtains

$$\dot{\boldsymbol{t}} = \boldsymbol{N}_l\boldsymbol{N}_d\ddot{\boldsymbol{q}} + \boldsymbol{N}_l\dot{\boldsymbol{N}}_d\dot{\boldsymbol{q}} + \dot{\boldsymbol{N}}_l\boldsymbol{N}_d\dot{\boldsymbol{q}}. \tag{30}$$

Then substituting (30) into (29), the independent set of constraint dynamic equations of motion for the flexible robot are obtained as

$$\boldsymbol{I}\ddot{\boldsymbol{q}} = \boldsymbol{\phi}, \tag{31}$$

where $\boldsymbol{I}$ is the $\bar{n} \times \bar{n}$ Generalized Inertia Matrix (GIM) of the flexible system. The (i, j) element of the matrix, $\boldsymbol{I}$, are the $(1 + 3m_i + \bar{m}_i) \times (1 + 3m_i + \bar{m}_i)$ block matrices, which can be expressed analytically as

$$\boldsymbol{I}_{ij} = \boldsymbol{I}_{ji}^{\mathrm{T}} = \boldsymbol{P}_i^{\mathrm{T}} \tilde{\boldsymbol{M}}_i \boldsymbol{A}_{ij} \boldsymbol{P}_j, \quad \text{for } i = 1, \ldots, n; \ \ j = 1, \ldots, i, \tag{32}$$

where the $(1 + 3m_i + \bar{m}_i) \times (1 + 3m_i + \bar{m}_i)$ matrix, $\tilde{\boldsymbol{M}}_i$, is obtained recursively as

$$\tilde{\boldsymbol{M}}_i \equiv \boldsymbol{M}_i + \boldsymbol{A}_{i+1,i}^{\mathrm{T}} \tilde{\boldsymbol{M}}_{i+1} \boldsymbol{A}_{i,i+1}$$

in which $\tilde{\boldsymbol{M}}_{n+1} \equiv \boldsymbol{O}$, as there is no $(n+1)$st link in the chain. Hence, $\tilde{\boldsymbol{M}}_n \equiv \boldsymbol{M}_n$. Moreover, the $\bar{n}$-dimensional vector, $\boldsymbol{\phi}$, containing external, strain energy, Coriolis, and other velocity dependent terms is expressed as

$$\boldsymbol{\phi} = \boldsymbol{\tau} - \boldsymbol{N}_d^{\mathrm{T}} \boldsymbol{N}_l^{\mathrm{T}} \big[\boldsymbol{M} \big(\boldsymbol{N}_l \dot{\boldsymbol{N}}_d + \dot{\boldsymbol{N}}_l \boldsymbol{N}_d \big) \dot{\boldsymbol{q}} \big]. \tag{33}$$

Equations (31)–(33) represent the analytical and recursive expressions for the matrix elements of the GIM. Similar to the rigid body system, [35], $\tilde{\boldsymbol{M}}_i$ is interpreted here as the mass matrix for the "composite flexible body" comprising of the rigidly attached flexible bodies $\#i, \ldots, \#n$. Note that except their dimensions, the expression of the DeNOC matrices and the recursive expressions associated with the dynamic models for the flexible systems are exactly the same as those for rigid link systems. This feature is exploited to build up a unified forward dynamics algorithm for the robots with both rigid and flexible links.

5 Forward dynamics algorithm

In this section, a recursive, computationally efficient, and numerically stable algorithm for calculating the joint accelerations, $\ddot{\boldsymbol{q}}$ from (31) are outlined. The GIM, $\boldsymbol{I}$ of (31), is decomposed using the reverse Gaussian elimination (RGE) [24, 38], namely,

$$\boldsymbol{I} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^{\mathrm{T}}, \tag{34}$$

where $\boldsymbol{U}$ and $\boldsymbol{D}$ are respectively the $\bar{n} \times \bar{n}$ upper block triangular and block diagonal matrices, and $\bar{n}$ is the degree of freedom of the system given by, $\bar{n} \equiv n + \sum_{i=1}^{n_f} (3m_i + \bar{m}_i)$. Matrices $\boldsymbol{U}$ and $\boldsymbol{D}$ are

$$\boldsymbol{U} \equiv \begin{bmatrix} \boldsymbol{1} & \boldsymbol{U}_{12} & \cdots & \boldsymbol{U}_{1n} \\ \boldsymbol{O} & \boldsymbol{1} & \cdots & \boldsymbol{U}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O} & \boldsymbol{O} & \cdots & \boldsymbol{1} \end{bmatrix}; \qquad \boldsymbol{D} \equiv \begin{bmatrix} \hat{\boldsymbol{I}}_1 & \boldsymbol{O} & \cdots & \boldsymbol{O} \\ \boldsymbol{O} & \hat{\boldsymbol{I}}_2 & \cdots & \boldsymbol{O} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O} & \cdots & \cdots & \hat{\boldsymbol{I}}_n \end{bmatrix}, \tag{35}$$

where the $(1 + 3m_i + \bar{m}_i) \times (1 + 3m_i + \bar{m}_i)$ matrices, $\boldsymbol{U}_{ij}$ and $\hat{\boldsymbol{I}}_i$, for $i, j = 1, \ldots, n$, are obtained from the application of the RGE rules. Note that for the rigid bodies, matrices $\boldsymbol{U}_{ij}$

and $\hat{\boldsymbol{I}}_i$ are scalars and $\boldsymbol{U}$ and $\boldsymbol{D}$ reduce to $n \times n$ upper triangular and diagonal matrices [37]. The expressions for $\boldsymbol{U}_{ij}$and $\hat{\boldsymbol{I}}_i$ for $i = 1, \ldots, n$ and $j = i + 1, \ldots, n$, are

$$\boldsymbol{U}_{ij} \equiv \boldsymbol{P}_i^{\mathrm{T}} \boldsymbol{\Psi}_{ij}, \quad \text{and} \quad \hat{\boldsymbol{I}}_i \equiv \boldsymbol{P}_i^{\mathrm{T}} \hat{\boldsymbol{\Psi}}_i, \tag{36}$$

where $\boldsymbol{P}_i$ is the $(6+3m_i+\bar{m}_i) \times (1+3m_i+\bar{m}_i)$-dimensional joint displacement amplitude (JDA) propagation matrix as defined in (9), and the $(6 + 3m_i + \bar{m}_i) \times (1 + 3m_i + \bar{m}_i)$ matrices, $\hat{\boldsymbol{\Psi}}_i$ and $\boldsymbol{\Psi}_{ij}$ are given by

$$\hat{\boldsymbol{\Psi}}_i \equiv \hat{\boldsymbol{M}}_i \boldsymbol{P}_i; \qquad \boldsymbol{\Psi}_i \equiv \hat{\boldsymbol{\Psi}}_i \hat{\boldsymbol{I}}_i^{-1}; \qquad \boldsymbol{\Psi}_{ij} \equiv \boldsymbol{A}_{ji}^{\mathrm{T}} \boldsymbol{\Psi}_i. \tag{37}$$

Again, for the rigid bodies, matrices $\hat{\boldsymbol{\Psi}}_i$ and $\boldsymbol{\Psi}_{ij}$ reduce to 6-dimensional vectors, whereas the $(6 + 3m_i + \bar{m}_i) \times (1 + 3m_i + \bar{m}_i)$ dimensional JDA propagation matrix, $\boldsymbol{P}_i$, reduces to the 6-dimensional vector, $\boldsymbol{p}_i$, defined in (10). Also, the $(6 + 3m_i + \bar{m}_i) \times (6 + 3m_i + \bar{m}_i)$ flexible twist propagation matrix, $\boldsymbol{A}_{ij}$, defined in (9), reduces to the 6×6 matrix, $\boldsymbol{R}_{ij}$ of (10). Note that the matrix, $\hat{\boldsymbol{M}}_i$, represents the mass and inertia properties of the articulated flexible body i, defined similar to the articulated inertia matrix for the rigid body system, consisting of flexible links, $\#i, \ldots, \#n$, which are coupled by joints $i + 1, \ldots, n$ [35]. The $(6 + 3m_i + \bar{m}_i) \times (6 + 3m_i + \bar{m}_i)$ matrix, $\hat{\boldsymbol{M}}_i$, is obtained recursively as

$$\hat{\boldsymbol{M}}_i \equiv \boldsymbol{M}_i + \boldsymbol{A}_{i,i+1}^{\mathrm{T}} \bar{\boldsymbol{M}}_{i+1} \boldsymbol{A}_{i,i+1}, \quad \text{where } \bar{\boldsymbol{M}}_i \equiv \hat{\boldsymbol{M}}_i - \hat{\boldsymbol{\Psi}}_i \boldsymbol{\Psi}_i^{\mathrm{T}} \tag{38}$$

for $i = n - 1, \ldots, 1$, and $\hat{\boldsymbol{M}}_n \equiv \boldsymbol{M}_n$. For the recursive algorithm, the $\bar{n}$-dimensional vector of the generalized forces due to external moments, forces, gravity, strain associated with the link deformations, and the centrifugal and Coriolis accelerations, etc., $\boldsymbol{\phi}$ of (31) and (33) are assumed to be evaluated recursively. The solution for $\ddot{\boldsymbol{q}}$ can now be obtained from (31) in the following three steps:

Step A. Solution for $\hat{\boldsymbol{\phi}}$*:* The solution, $\hat{\boldsymbol{\phi}} = \boldsymbol{U}^{-1}\boldsymbol{\phi}$, is evaluated for, $i = n - 1, \ldots, 1$ as

$$\hat{\boldsymbol{\phi}}_i = \boldsymbol{\phi}_i - \boldsymbol{P}_i^{\mathrm{T}} \boldsymbol{\eta}_i, \tag{39}$$

where $\hat{\boldsymbol{\phi}}_n \equiv \boldsymbol{\phi}_n$. Note that the $(6 + 3m_i + \bar{m}_i)$-dimensional vector, $\boldsymbol{\eta}_i$, is obtained as

$$\boldsymbol{\eta}_i = \boldsymbol{\Psi}_i \hat{\boldsymbol{\phi}}_i + \bar{\boldsymbol{\eta}}_i; \quad \bar{\boldsymbol{\eta}}_i \equiv \boldsymbol{A}_{i+1,i}^{\mathrm{T}} \boldsymbol{\eta}_{i+1}, \quad \text{and} \quad \bar{\boldsymbol{\eta}}_n = \boldsymbol{0}.$$

Step B. Solution for $\tilde{\boldsymbol{\phi}}$*:* $\boldsymbol{D}\tilde{\boldsymbol{\phi}} = \hat{\boldsymbol{\phi}}$, for $i = 1, \ldots, n$,

$$\tilde{\boldsymbol{\phi}}_i = \hat{\boldsymbol{I}}_i^{-1} \boldsymbol{\phi}_i \tag{40}$$

in which the $(1 + 3m_i + \bar{m}_i) \times (1 + 3m_i + \bar{m}_i)$ matrix $\hat{\boldsymbol{I}}_i$ is defined in (36). For the rigid body system, $\hat{\boldsymbol{I}}_i$ becomes scalar and $\hat{\boldsymbol{I}}_i^{-1}$ simply requires one division.

Step C. Solution for $\ddot{\boldsymbol{q}}_i$*:* $\ddot{\boldsymbol{q}}_i \equiv \boldsymbol{U}^{-\mathrm{T}}\tilde{\boldsymbol{\phi}}$, for $i = 2, \ldots, n$,

$$\ddot{\boldsymbol{q}}_i = \tilde{\boldsymbol{\phi}}_i - \boldsymbol{\Psi}_i^{\mathrm{T}} \bar{\boldsymbol{\mu}}_i, \tag{41}$$

where the $(1+3m_i+\bar{m}_i)$-dimensional vector, $\ddot{\boldsymbol{q}}_1 \equiv \tilde{\boldsymbol{\phi}}_1$, and the $(6+3m_i+\bar{m}_i)$-dimensional vector, $\bar{\boldsymbol{\mu}}_i$, is obtained from

$$\bar{\boldsymbol{\mu}}_i \equiv \boldsymbol{A}_{i,i-1} \boldsymbol{\mu}_{i-1} \quad \text{and} \quad \boldsymbol{\mu}_i \equiv \boldsymbol{P}_i \ddot{\boldsymbol{q}}_i + \bar{\boldsymbol{\mu}}_i; \quad \bar{\boldsymbol{\mu}}_1 = \boldsymbol{0}. \tag{42}$$

For the rigid bodies, $\boldsymbol{A}_{i,i-1} \equiv \boldsymbol{R}_{i,i-1}$, $\boldsymbol{P}_i \equiv \boldsymbol{p}_i$, and $\bar{\boldsymbol{\mu}}_i$ is a 6-dimensional vector.

5.1 Computational complexity

Using the above steps, namely, in A, B, C, above, it can be seen that unlike the rigid-body system where the equivalence of $\hat{\boldsymbol{I}}$ is a scalar [37], for flexible bodies it is the $(1+\tilde{m}) \times (1+\tilde{m})$ matrix. Note here that for the estimation of computational complexity, all links are assumed to vibrate in the same number of modes, i.e., m_i and $\bar{m}_i = \bar{m}$. Accordingly, $\tilde{m}$ is taken as, $\tilde{m} = 3m + \bar{m}$. Since the individual element of the matrix, $\hat{\boldsymbol{I}}_i$, do not have any recursive relation like the block matrix elements of $\boldsymbol{I}$, no analytical decomposition is possible. Hence, $\hat{\boldsymbol{I}}_i^{-1}$ requires explicit inversion whose computational complexity is in the order of $\frac{(1+\tilde{m})^3}{3} \approx O(\frac{\tilde{m}^3}{3})$, as m is generally large. However, the rest of the calculations can be done with $O(n)$—n being the total number of links—calculations. In order to justify this, let us assume that (39) needs C_1 computations. Hence, Step A requires nC_1 calculations. Next, (40) will require C_2 the computations, which includes $\hat{\boldsymbol{I}}_i^{-1}$. Hence, the order of computations for C_2 is $\tilde{m}^3/3$ as mentioned above, and computations required in Step B is nC_2. Finally, if (41) and (42) require C_3 computations, then Step C can be obtained with nC_3 calculations. So, the total computations to solve for the accelerations of the generalized coordinates, $\ddot{\boldsymbol{q}}$, are $n(C_1 + C_2 + C_3)$, i.e., the order of computation is $O(n) + O(\frac{\tilde{m}^3}{3})$. It is pointed out here that if the conventional formulation like the one reported in [13] and others are used; where explicit inversion of $\boldsymbol{I}$, (31), is carried out, the order of computations would be $O(\frac{\bar{n}^3}{3})$—$\bar{n}$ being equal to $n_r + (\tilde{m}+1)n_f$.

Next, the exact number of computations required in the proposed forward dynamics algorithm are counted. In the count, the effect of payload and hub are not considered so as to be able to compare the final complexity with those reported in the literature, where these effects are also not reported. The total computational counts are given as

$$\begin{aligned}
&\left[\left(2m^2 + \bar{m}^2 + 180m + 4\bar{m} + 299 + M_{m1}\right)n - \left(124m + \bar{m} + 229 + M_{m2}\right)\right] \quad \text{M;}\\
&\left[\left(6m^2 + 2\bar{m}^2 + 4m\bar{m} + 122m + 8\bar{m} + 220 + A_{m1}\right)n - \left(94m + \bar{m} + 192 + A_{m2}\right)\right] \quad \text{A;}\\
&\text{and } 2n \text{ T,}
\end{aligned} \tag{43}$$

where M_{m1}, M_{m2}, A_{m1} and A_{m2} are the functions of $\hat{m} = 1 + 2m + \bar{m}$, and depend on the number of modes considered for the vibrations. They are given by

$$\begin{aligned}
M_{m1} &\equiv \frac{\hat{m}^3}{6} + \left(2m + \bar{m} + \frac{15}{2}\right)\hat{m}^2 + \left(\frac{4m^2 + \bar{m}^2 + 4m\bar{m} + 30m + 15\bar{m}}{2} + \frac{100}{3}\right)\hat{m};\\
M_{m2} &\equiv \left(27 + \frac{4m^2 + \bar{m}^2 + 4m\bar{m} + 26m + 13\bar{m}}{2}\right)\hat{m};\\
A_{m1} &\equiv \frac{\hat{m}^3}{6} + \frac{\hat{m}^2}{2} + \left(6m^2 + \frac{3}{2}\bar{m}^2 + 6m\bar{m} + 27m + \frac{27}{2}\bar{m} + \frac{82}{3}\right)\hat{m};\\
A_{m2} &\equiv \left(2m^2 + \frac{\bar{m}^2}{2} + 2m\bar{m} + 13m + \frac{13}{2}\bar{m} + 27\right)\hat{m}.
\end{aligned}$$

5.2 Comparison

The computational complexity of the proposed algorithm is compared in Table 1 with the two other algorithms available in the literature, namely, [13, 46]. No other recent literature

Table 1 Comparison of computational complexity

Algorithm	Multiplications (M)	Additions (A)	M	A
(**a**) Considering vibrations in bending ($m = 2$) and torsion ($\bar{m} = 1$)				
Proposed ($n = 6$)[a]	$M_{1pt}n + M_{2pt}$	$A_{1pt}n + A_{2pt}$	**9038**	**7311**
Cyril [13] ($n = 6$)[b]	$M_{1c}n^3 + M_{2c}n^2 + M_{3c}n + M_{4c}$	$A_{1c}n^3 + A_{2c}n^2 + A_{3c}n + A_{4c}$	10812	8538
(**b**) Considering vibrations in bending only ($m = 2$)				
Proposed ($n = 8$)[a]	$M_{1p}n + M_{2p}$	$A_{1p}n + A_{2p}$	**9874**	**7699**
Jain and Rodriguez[b] [23, 24] ($n = 9$)[a]	$M_{1j}n$	$A_{1j}n$	11889	9387
Jain and Rodriguez[c] [23, 24]	$M^{nr}_{1j}n^3 + M^{nr}_{2j}n^2 + M^{nr}_{3j}n$	$A^{nr}_{1j}n^3 + A^{nr}_{2j}n^2 + A^{nr}_{3j}n$	9920 ($n = 8$) 12360 ($n = 9$)	8605 ($n = 8$) 10660 ($n = 9$)

[a]Number of links for which the recursive algorithms benefit over non-recursive algorithm appearing in that table

[b]Recursive algorithm

[c]Non-recursive algorithm

reported such complexity count except for the RR, where R stands for revolute joints, and RRR robots [49]. In [49], only the computational complexity for the generalized inertia matrix (GIM) is reported, but not for the forward dynamics algorithm. This may be apparently due to the complexity in counting the number of operations in terms of multiplications, additions, etc. However, it is emphasized here due to the fact that even a few savings in multiplications or additions may have significant impact on the efficiency of the algorithm when these computations are performed over thousands and millions of discretized steps. As a result, one can apply the algorithm in real-time applications or use less advanced, less costly computing system to perform the same work satisfactorily. Note that the algorithm and the computational count given by Cyril [13] includes the effect of torsional vibrations, whereas the algorithm and the computational count given by Jain and Rodriguez [22] does not consider them. Hence, the proposed algorithm is compared with the above two algorithms separately in Table 1. From Tables 1(a–b), it is clear that the recursive algorithm proposed above outperforms the other two.

The coefficients in Tables 1(a) and 1(b) are:

$$
\begin{aligned}
M_{1pt} &\equiv 2m^2 + \bar{m}^2 + 180m + 4\bar{m} + 299 + M_{m1};\\
M_{2pt} &\equiv -\big(124m + \bar{m} + 229 + M_{m2}\big);\\
A_{1pt} &\equiv 6m^2 + 2\bar{m}^2 + 4m\bar{m} + 122m + 8\bar{m} + 220 + A_{m1};\\
A_{2pt} &\equiv -\big(94m + \bar{m} + 192 + A_{m2}\big);
\end{aligned}
$$

where M_{m1}, M_{m2}, A_{m1}, and A_{m2} are given after (43) and

$$
M_{1c} \equiv \frac{\hat{m}^3}{6} + \hat{m}^2\left(-\frac{m}{2} + 1\right) + \hat{m}\left(\frac{m^2}{2} - 2m\right) - \frac{m^3}{6} + m^2;
$$

$$M_{2c} \equiv -\frac{3\hat{m}^2}{2} + \hat{m}\left(3m + \frac{33}{2}\right) - \frac{3m^2}{2} - \frac{15m}{2} + \frac{15\bar{m}}{2} + 12;$$

$$M_{3c} \equiv 5\hat{m}^2 - \hat{m}\left(10m + \frac{109}{6}\right) + \frac{m^3}{6} + \frac{53m^2}{6} + \frac{233m}{3} - \frac{7\bar{m}}{2} + 246;$$

$$M_{4c} \equiv -123;$$

$$A_{1c} \equiv \frac{\hat{m}^3}{6} + \hat{m}^2\left(-\frac{m}{2} + 1\right) + \hat{m}\left(\frac{m^2}{2} - 2m\right) - \frac{m^3}{6} + m^2;$$

$$A_{2c} \equiv -\frac{3\hat{m}^2}{2} + \hat{m}(3m + 13) - \frac{3m^2}{2} - \frac{17m}{2} + 3\bar{m} + 9;$$

$$A_{3c} \equiv 5\hat{m}^2 - \hat{m}\left(10m + \frac{85}{6}\right) + \frac{m^3}{6} + \frac{53m^2}{6} - \frac{251m}{6} + 228;$$

$$A_{4c} \equiv -102;$$

$$M_{1p} \equiv 2m^2 + 180m + 299 + M_{m3};$$

$$M_{2p} \equiv -(124m + 229 + M_{m4});$$

$$A_{1p} \equiv 6m^2 + 122m + 220 + A_{m3};$$

$$A_{2p} \equiv -(94m + 192 + A_{m4});$$

where

$$M_{m3} \equiv \frac{\hat{m}^3}{6} + \left(2m + \frac{15}{2}\right)\hat{m}^2 + \left(2m^2 + 15m + \frac{100}{3}\right)\hat{m};$$

$$M_{m4} \equiv \left(2m^2 + 13m + 27\right)\hat{m};$$

$$A_{m3} \equiv \frac{\hat{m}^3}{6} + \frac{\hat{m}^2}{2} + \left(6m^2 + 27m + \frac{82}{3}\right)\hat{m};$$

$$A_{m4} \equiv \left(2m^2 + 13m + 27\right)\hat{m};$$

$$M_{1j} \equiv \begin{cases} \frac{5m^3}{6} + \frac{27m^2}{2} + \frac{893m}{3} + 281 & \text{for } n \le 7, \\ 13m^2 + 298m + 673 & \text{for } n \ge 7; \end{cases}$$

$$A_{1j} \equiv \begin{cases} \frac{5m^3}{6} + \frac{23m^2}{2} + \frac{677m}{3} + 256 & \text{for } n \le 7, \\ 14m^2 + 225m + 537 & \text{for } n \ge 7; \end{cases}$$

$$M_{1j}^{nr} \equiv \frac{1}{6}; \qquad M_{2j}^{nr} \equiv 12m^2 + 34m + \frac{29}{2};$$

$$M_{3j}^{nr} \equiv 48m + \frac{268}{3};$$

$$A_{1j}^{nr} \equiv \frac{1}{6}; \qquad A_{2j}^{nr} \equiv 11m^2 + 24m + 14;$$

$$A_{3j}^{nr} \equiv m^2 + \frac{97}{2}m + 116.$$

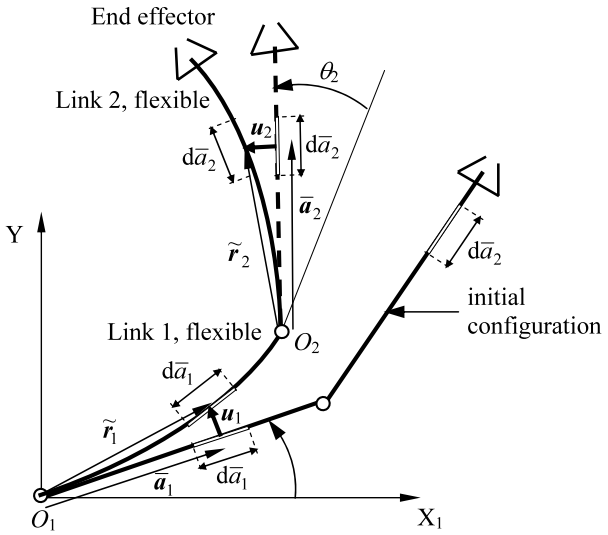


Fig. 3 A two-link arm with both links flexible

Note that from the computational point of view using the nonrecursive algorithms for $n < 8$, where n is the number of links in the robotic chain is economical as seen from the Table 1(b). However, due to the nature of nonrecursive algorithms the numerical methods employed to obtain the joint velocities and positions do not converge as quickly as for those obtained using a recursive algorithm. Hence, overall CPU time to calculate the joint accelerations from forward dynamics, followed by numerical integration of the joint accelerations using a nonrecursive algorithm may be more. This nonconvergence of joint accelerations in case of nonrecursive algorithms also leads to numerical instability in the simulation results. This aspect is explained in Sect. 7. Hence, a recursive algorithm is preferred over nonrecursive algorithms in forward dynamics and simulation.

6 Simulations

In order to validate the proposed forward dynamics algorithm, simulation of a two-link flexible robotic arm and a spatial 6-link Space Shuttle Remote Manipulator System Robot (SSRMS), with its 2nd and 3rd links flexible are performed. In both the cases, the flexible links are assumed to be vibrating in their first two modes only. The eigen function used to represent the shape functions of the ith link in its jth mode of bending is given by [26]:

$$\begin{aligned} s_{ij} &= \left[(\sin\varsigma_j + \sinh\varsigma_j) - \sigma(\cos\varsigma_j + \cosh\varsigma_j)\right]; \\ \sigma &= (\sin\delta_j + \sinh\delta_j)/(\cos\delta_j + \cosh\delta_j), \end{aligned} \tag{44}$$

where for $j = 1$ and 2, the modal constant, $\delta_j = 1.875$ and 4.694, respectively [26]. The shape functions of the link in torsion $\bar{s}_{ij}$, as defined in (3b), can also be represented by similar trigonometric functions [30].

6.1 Both links flexible

Simulation of a two-link arm with both links flexible, as shown in Fig. 3 is performed. The arm is considered hanging freely under gravity and no external torques are applied on it.

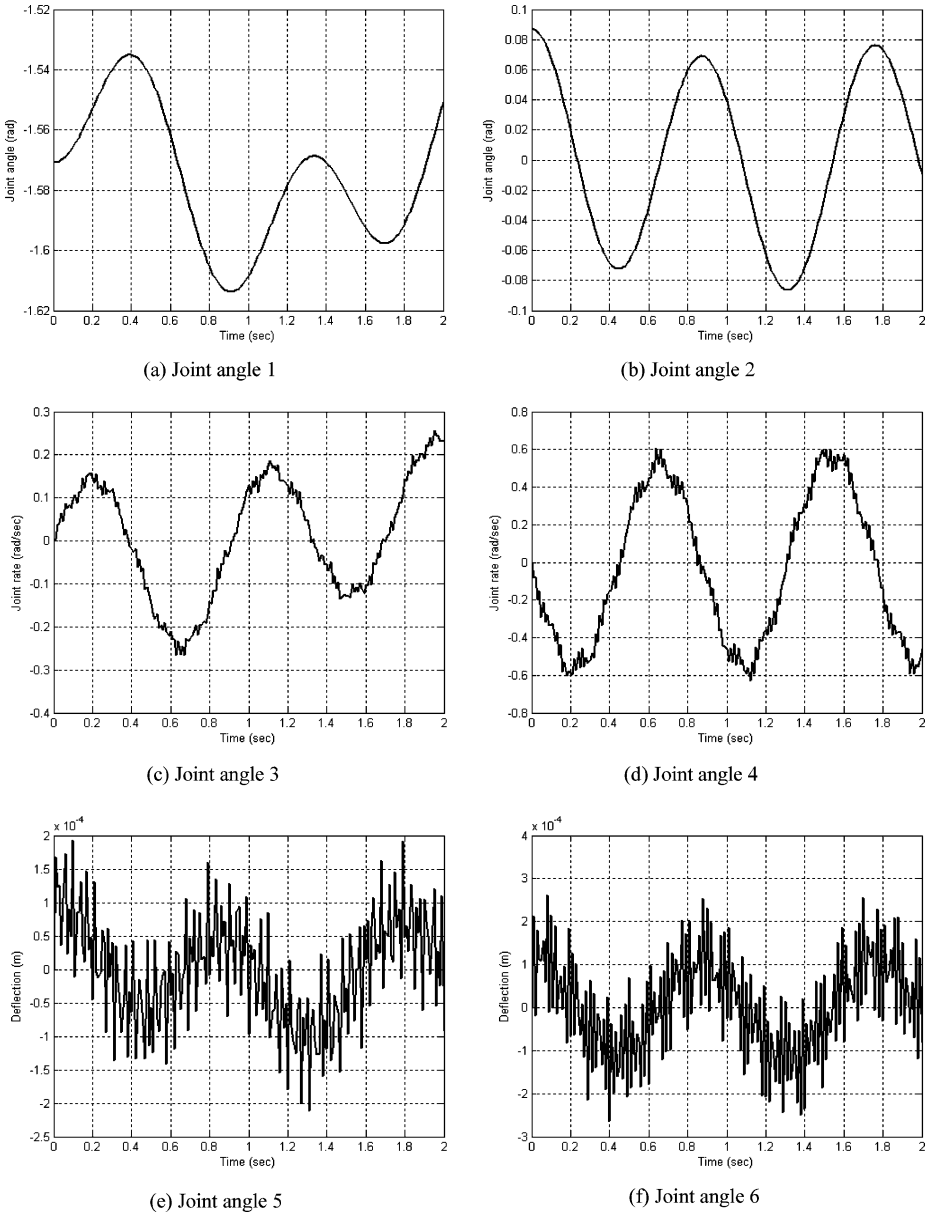


Fig. 4 Simulation results for two-link flexible arm

Both the links have mass, length, and flexure rigidity of 1 m, 5 Kg and 1000 Nm^2. The response of the system is obtained with the following initial conditions: $\theta_1 = -90°$, $\theta_2 = 5°$, $\dot{\theta}_1 = 0$ and $\dot{\theta}_2 = 0$, and $d_{i,j} = 0$ and $\dot{d}_{i,j} = 0$, for $i, j = 1, 2$. Simulation results, shown in Fig. 4, match exactly with those given in the literature, namely, in [13]. It is pointed out here that Cyril [13] had observed artificial damping in his simulation results, which are absent using the present formulation. This can be attributed to the numerical stiffness present in the

original-NOC based nonrecursive formulation proposed by Cyril [13]. The present DeNOC based recursive algorithm avoids such artificial stiffness. These aspects are elaborated in Sect. 7.

6.2 Space shuttle remote manipulator system robot

Next, the 6-link Space Shuttle Remote Manipulator System (SSRMS) [13, 45] is considered, whose 2nd and 3rd links are assumed flexible due to its architecture. Figure 5 shows the SSRMS whose DH and other parameters are given in Table 2. Forced simulation is performed using the scheme outlined in Fig. 6. The joint trajectories, given by (45), are prescribed for a representative maneuver of the SSRMS, considering all links are rigid. The joint torques are then computed using the inverse dynamics algorithm, developed separately, for the SSRMS robot, considering all of its links as rigid. The joint torques thus obtained are shown in Fig. 7

$$\theta_i = 0.05\left[t - \frac{5}{\pi}\sin\left(\frac{\pi}{5}t\right)\right], \quad \text{for } i = 1, \ldots, 6. \tag{45}$$

The simulation results are given in Figs. 8–9. The results match with those presented in [13]. The tip deflection of the flexible links 2 and 3 are shown in Fig. 9. The simulation results obtained for the robot are stable up to about 6 sec. After this numerical errors start building up resulting in unstable results. These aspects are separately discussed in Sect. 7.

Table 2 DH and other parameters of the SSRMS robot

Link No. (i)	a_i (m)	b_i (m)	α_i (rad)	θ_i (rad)	m_i (Kg)	$E_i I_i$ (Nm^2)
1	1	0	$-\pi/2$	$\theta_1[0]$[a]	47.5	–
2	6	0	0	$\theta_2[0]$	140	1×10^5
3	7	0	$\pi/2$	$\theta_3[0]$	85	1×10^5
4	1	0	$-\pi/2$	$\theta_4[0]$	47.5	–
5	1	0	$\pi/2$	$\theta_5[0]$	47.5	–
6	1	0	0	$\theta_6[0]$	47.5	–

[a]The values in $[\cdot]$ show the initial configuration of the arm

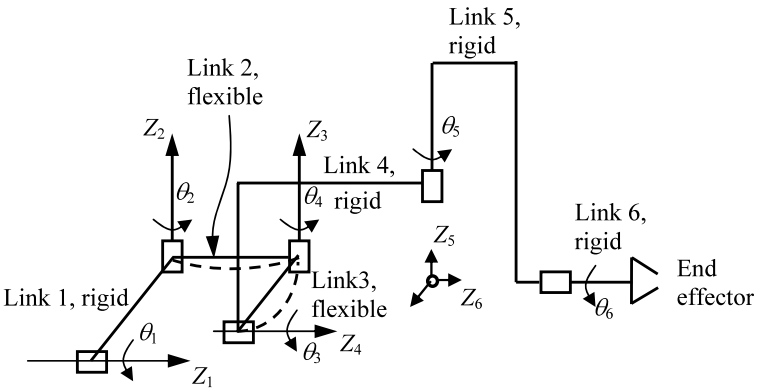


Fig. 5 The six-link SSRMS

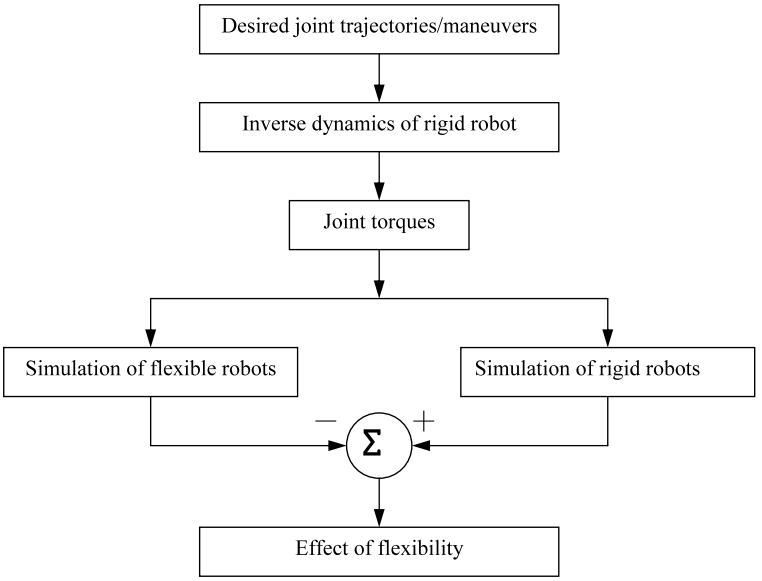


Fig. 6 Forward dynamics analysis of SSRMS

However, up to 6 sec. whatever variations are noted, they are due to the flexibility of the links.

7 Numerical stability and CPU time

In Sect. 5, the computational complexity of the forward dynamics algorithm proposed in the paper is shown to be better than some other algorithms reported in the literature. However, the computational complexity of the forward dynamics cannot alone be considered for the overall efficiency, as the integration scheme and the numerical stability characteristics of the forward dynamics algorithm also play important roles. Assuming an appropriate numerical integration scheme is used, the numerical stability of the proposed algorithm and the associated CPU time are studied in this section.

7.1 Numerical stability

The numerical stability of the proposed algorithm is studied here taking the numerical example of the simulations of SSRMS robot, considered in Sect. 6.2. The simulation results are obtained using the ode45 function with 0.001-step size and 10^{-8} tolerance in Matlab v6.5. The simulation results obtained using the proposed algorithm are compared with those obtained using a non-recursive algorithm while all other calculations strategies are kept the same. In the nonrecursive algorithm, the GIM, $\boldsymbol{I}$ of (31), is first obtained numerically before it is factorized numerically using the Cholesky decomposition [48], followed by the solution of the joint accelerations q, using forward and backward substitutions. The algorithm is known to have $O(\frac{\bar{n}^3}{3})$ complexities [48]. Note that for the nonrecursive algorithms, as described in Sect. 1, there exist a number of stabilization methods, namely, Baumgarte

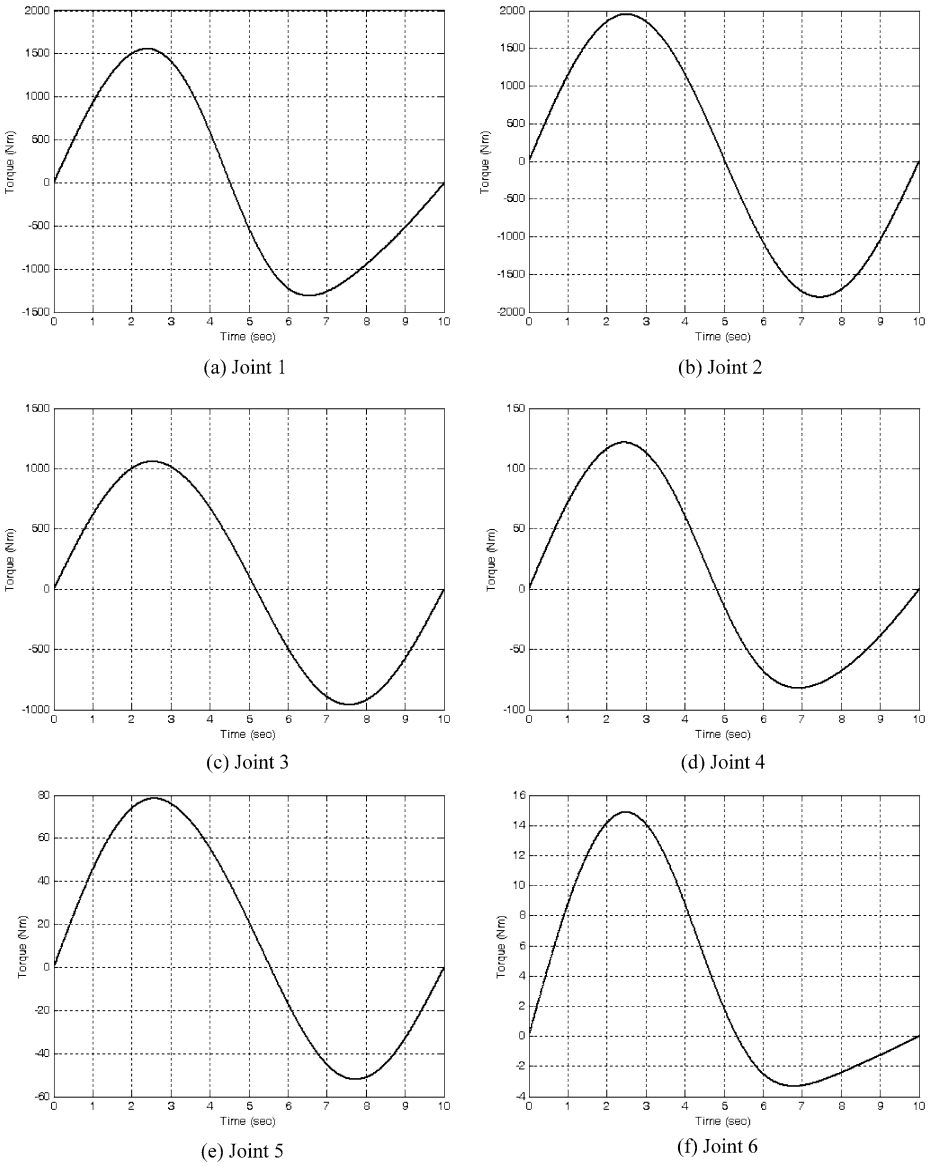


Fig. 7 Joint torques for SSRMS

stabilization and augmented Lagrangian approach [32]. However, these stabilization techniques modify the dynamic model such that the simulation results obtained do not correspond to the original system, but represent some other slightly deviated system [40]. Moreover, behavior of these stabilization approaches is highly configuration dependent and does not improve the stability of the system for its near singular configurations [33]. Furthermore, the above stabilization techniques have been used by researchers for only rigid link robotic systems. More recently, researchers have proposed methods for solving efficiently systems of differential-algebraic equations which represent flexible mechanisms. However, stability

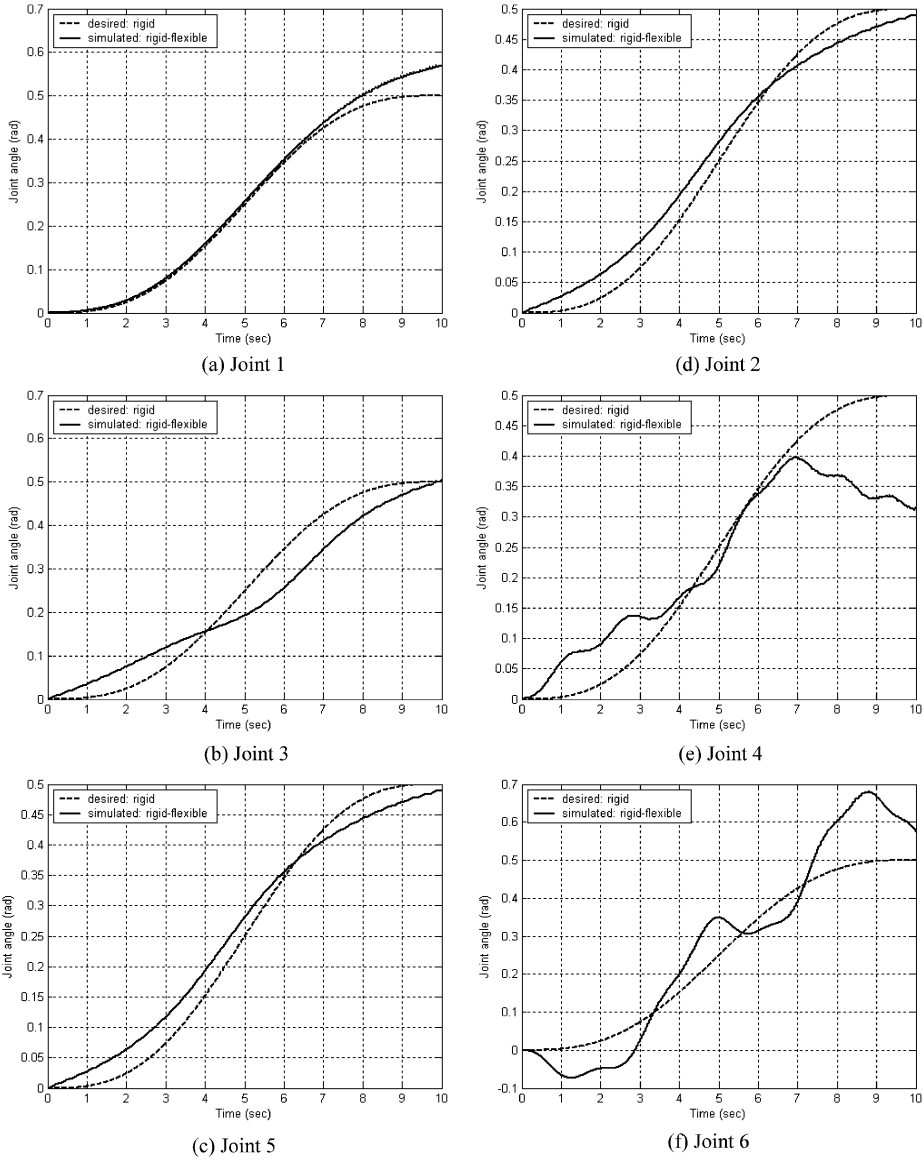
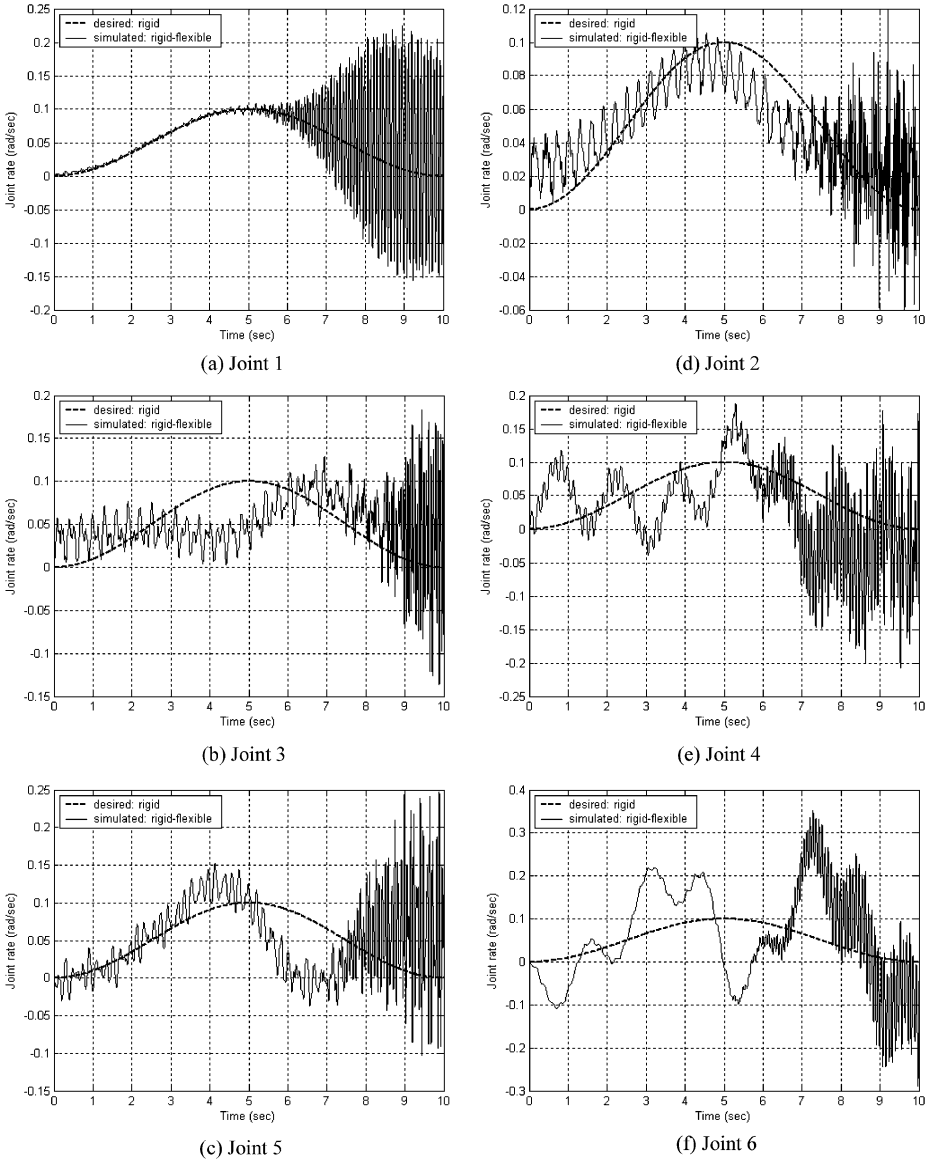


Fig. 8 Comparison of joint responses of SSRMS

analysis of algorithms for rigid-flexible link robotic systems is still an open area of research. In order to avoid the loss in accuracy of simulation results, no stabilization method is used here with any of the algorithms. The calculations of $\boldsymbol{\phi}$ are carried out exactly in a manner done for the proposed recursive algorithm. Hence, the effect of recursive and nonrecursive algorithms becomes explicit. The simulation results obtained using the two algorithms are compared with the desired trajectory of (45). The difference between the simulated joint path obtained from the two FD algorithms and the desired joint maneuver are plotted in



(a) Joint 1 (d) Joint 2

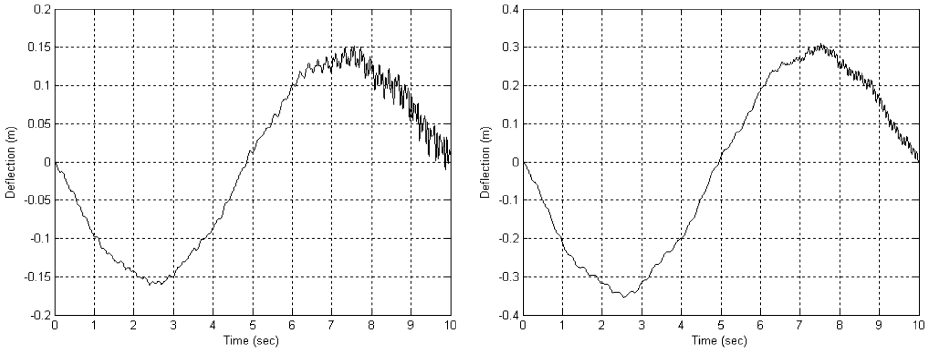
(b) Joint 3 (e) Joint 4

(c) Joint 5 (f) Joint 6

(**b**) Comparison of joint rates

Fig. 8 (*Continued*)

Fig. 10 for joint 4. Comparison is shown for the results of joint 4 only, as the effect of numerical stability is most pronounced. Similar behavior was obtained for the other joints, and hence not reported here. It is seen that the simulation based on the nonrecursive algorithm totally fails after about 1.2 sec. On the other hand, the simulation using the proposed recursive algorithm continues up to 2 sec. and up to 7 sec. in total, albeit the error blows up. Next, the stability of the two algorithms is investigated using the criterion based on the principle of conservation of energy and power as proposed by Sharf and Damaren [45]. In



(a) Link 1 (b) Link 2

Fig. 9 Tip deflection of the flexible links of SSRMS

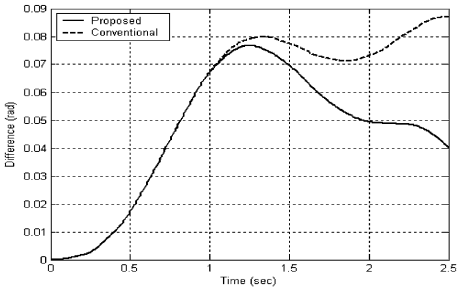


Fig. 10 Error deviations for joint 4 of SSRMS

the forced simulation, the input power to the robotic system is calculated as, $\Pi = \sum_{i=1}^{n} \tau_i \dot{\theta}_i$, where τ_i and $\dot{\theta}_i$ are the torque obtained from the inverse dynamics algorithm and the desired joint rates, respectively, for the all rigid SSRMS system-n being the number of joints. Now, based on the simulation results, the simulated power is computed as $\tilde{\Pi} = \sum_{i=1}^{n} \tau_i \dot{\tilde{\theta}}_i$, where $\dot{\tilde{\theta}}_i$ is the simulated joint rate. Since a flexible link is energy dissipating system, assuming no losses due to friction and damping, the output energy is equal to the input energy supplied to the system minus the energy dissipated due to vibrations. Thus, to compare the numerical stability of the two algorithms, namely, the proposed and the non-recursive algorithm, the output power is calculated from both the algorithms. The results are plotted in Fig. 11. For the nonrecursive algorithm, results become unstable and simulation stops soon after 2.5 sec. However, for the recursive algorithm, the output power almost matches with the input one even up to 6 sec. Moreover, the drift in the nonrecursive algorithm increases after about 1.1 sec. before it fails after 2.5 sec. The drift for the recursive algorithm is much less even up to 6 sec. Hence, the numerical stability of the latter algorithm is established. Next, in order to investigate the built-up of the errors, the nature of the joint accelerations, $\dot{\theta}_i$, for $i = 1, \ldots, 6$, is obtained using both the algorithms. Figure 12 plots the joint acceleration for joint 6, which also shows the better performance for the proposed recursive algorithm. Similar behavior is also observed for other joint accelerations.

It should be noted here that the formulation, as presented in Sect. 4.2, derives a set of independent dynamic equations of motion, which are Ordinary Differential Equations (ODE) and the ODEs are known to provide numerically stable algorithms compared to the Differ-

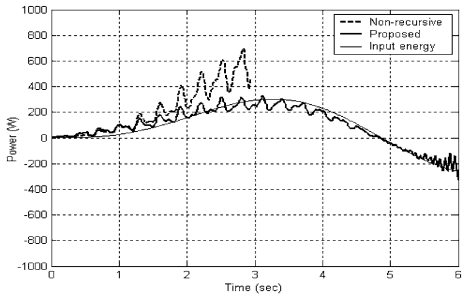
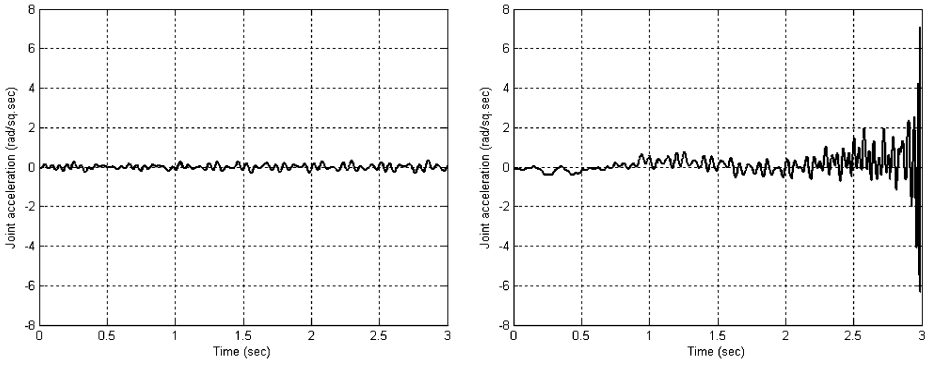


Fig. 11 Comparison of desired and simulated powers



(a) Recursive Proposed (b) Non recursive

Fig. 12 Joint accelerations of joint 6 of SSRMS

ential Algebraic Equations (DAE) representing the same system dynamics [33]. There is as such no algebraic constraint in the proposed formulation of the system of dynamic equation of motion, namely, (31). Hence, the integrator ode45 used in simulation need not handle any algebraic equations. Moreover, to check if the numerical stiffness due to the link flexibility has affected the results or not, the same set of simulations are carried out using ode23s as the numerical integrator in Matlab v. 2007a. Note that the ode23s integrator can handle stiff systems efficiently. The results obtained using ode45 and ode23s match exactly. Thus, the apparent differences in the characteristics of the simulation results are only due to numerical stability caused by truncation error, round-off error, etc. in the algorithms.

7.2 CPU time

To investigate the efficiency of the proposed algorithm, the CPU times taken by the recursive and the nonrecursive algorithms for the forced simulation of the SSRMS robot arm is obtained. The CPU times taken by the 3.5 GHz Pentium PC using the two algorithms for the simulation duration of 2.5 sec. is noted down, which are shown in Fig. 13. The CPU times were obtained using "tic" and "toc" commands of the MATLAB at the beginning and the end of the program for the simulation. The reason for taking the simulation time of 2.5 sec. is because up to 2.5 sec. both the algorithms give stable results. It is clear from Fig. 13 that the proposed $O(n)$ recursive algorithm is more efficient than its nonrecursive counterpart. This is due to the fact that the joint accelerations, $\dot{\theta}_i$ for $i = 1, \ldots, 6$, obtained for the recursive algorithm are smoother for a longer duration of time than the nonrecursive one, as

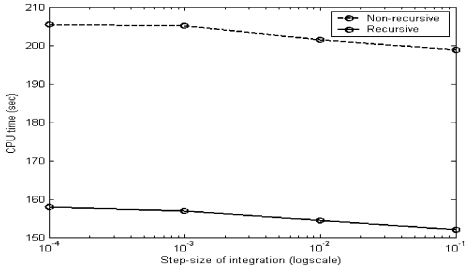


Fig. 13 Comparison of CPU times

seen for joint 6 in Fig. 12. Other joints behave similarly. Because of smooth nature of $\dot{\theta}_i$, the numbers of iterations required in the numerical integration are less, hence requiring less CPU time for the proposed recursive algorithm. These results are also consistent with the power plots shown in Fig. 11. In order to see the effect of the step sizes on the CPU time taken by the recursive and the nonrecursive algorithms, step-sizes of 0.1, 0.01, 0.001, and 0.0001 are used in "ode45." No significant effect was observed.

8 Conclusions

A dynamic modeling approach for the serial-chain robots with flexible links based on the equivalence of Euler–Lagrange and Newton–Euler equations of motion, and the decoupled natural orthogonal complement (DeNOC) matrices is proposed that leads to a recursive forward dynamics algorithm. Such algorithm provides numerically stable and efficient simulation. The recursiveness was obtained due to the $\boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^{\mathrm{T}}$ decomposition of the generalized inertia matrix, (GIM). The decomposition is possible because of the decoupling of the NOC matrix. The proposed algorithm is shown to be computationally faster and numerically more stable. The contributions of this paper are:

(1) Simplification of the dynamic algorithm based on the assumption of link shapes as bent slender beams, which is realistic in most practical robot architectures; (2) Derivation and introduction of the DeNOC matrices in the dynamic modeling of flexible robots; (3) Evaluation and comparison of the computational complexity of the proposed recursive forward dynamics algorithm with those available in the literature; (4) Numerical stability analyses and efficiency of the proposed recursive forward dynamics algorithm based simulation of SSRMS. As per the authors' knowledge, such study is reported for the first time in the literature; and (5) Physical interpretations of many terms associated with the dynamic model of the flexible-link robots.

Acknowledgement The research work reported in this paper is carried out under the partial financial aid from the Department of Science and Technology, Government of India (SR/S3/RM/46/2002), which is duly acknowledged.

Appendix A: Denavit and Hartenberg parameters

The Denavit and Hartenberg (DH) parameters [16] are a systematic method to define the relative position and orientation of the consecutive links in a multibody robotic system and can be assigned differently for the same system, as in [31, 36]. The DH parameters that are used in this paper are explained here. Referring to Fig. 1, the serial robot manipulator under

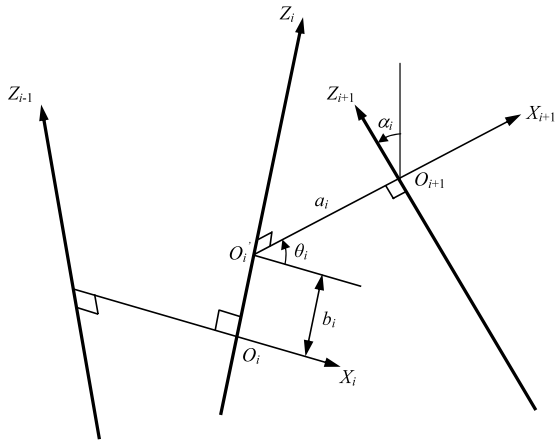


Fig. 14 Definition of DH parameters

study consists of $(n+1)$ bodies or links, namely, the fixed base and the bodies numbered as #1, ..., #n. All the bodies are coupled by n joints, numbered as $1, \ldots, n$. The ith joint couples the $(i-1)$st link; for the $(i+1)$st frame, i.e., X_{i+1}, Y_{i+1}, Z_{i+1}. Now referring to the Fig. 14 for the first n frames, the DH parameters are defined according to the following rules:

1. Z_i is the axis of the ith joint. Its positive direction can be chosen arbitrarily.
2. X_i is defined as the common perpendicular to Z_{i-1} and Z_i, directed from the former to latter. The origin of the ith frame, O_i, is the point where X_i intersects Z_i. If these two axes intersect, the positive direction of X_i is chosen arbitrarily. And the origin, O_i, coincides with the origin of the $(i-1)$st frame, i.e., O_{i-1}.
3. The distance between Z_i and Z_{i+1} is defined as a_i, which is a nonnegative number.
4. The Z_i coordinate of the intersection of the X_{i+1} axis with Z_i, which is shown in Fig. 14 as the distance between O_i and O_i' is defined as b_i. This can be either positive or negative. For a prismatic joint, b_i is a variable.
5. The angle between Z_i and Z_{i+1} is defined as α_i, and is measured about the positive direction of X_{i+1}.
6. The angle between X_i and X_{i+1} is defined as θ_i, and is measured about the positive direction of Z_i. For a revolute joint, θ_i is a variable.

Since no $(n+1)$st link exists, the above definitions do not apply to the $(n+1)$st frame and its axes can be chosen at will.

Appendix B: Derivation of (21)

Using the expressions for the total kinetic energy, $T = \sum_{i=1}^{n} T_i$, where T_i is given by (17), the partial differentiations with respect to the set of jth independent generalized speeds and coordinates, $\dot{\boldsymbol{q}}_j$ and $\boldsymbol{q}_j$, respectively, as required in (20) are obtained as

$$\frac{\partial T}{\partial \dot{\boldsymbol{q}}_j} = \sum_{i=1}^{n}\Bigg[\int_0^{b_i} \rho_i \dot{\boldsymbol{r}}_i^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{r}}_i}{\partial \dot{\boldsymbol{q}}_j}\,\mathrm{d}\bar{b}_i + \int_0^{a_i} \rho_i \dot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}} \frac{\partial \dot{\tilde{\boldsymbol{r}}}_i}{\partial \dot{\boldsymbol{q}}_j}\,\mathrm{d}\bar{a}_i + m_{pi} \dot{\boldsymbol{r}}_{pi}^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{r}}_{pi}}{\partial \dot{\boldsymbol{q}}_j} + \int_0^{a_i} \rho_i I_{pi} \dot{\beta}_i \frac{\partial \dot{\beta}_i}{\partial \dot{\boldsymbol{q}}_j}\,\mathrm{d}\bar{a}_i + (\boldsymbol{I}_{hi}\boldsymbol{\omega}_i)^{\mathrm{T}} \frac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j}\Bigg], \tag{B.1a}$$

$$\frac{\partial T}{\partial \boldsymbol{q}_j} = \sum_{i=1}^{n}\Bigg[\int_0^{b_i} \rho_i \dot{\boldsymbol{r}}_i^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{r}}_i}{\partial \boldsymbol{q}_j}\,\mathrm{d}\bar{b}_i + \int_0^{a_i} \rho_i \dot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}} \frac{\partial \dot{\tilde{\boldsymbol{r}}}_i}{\partial \boldsymbol{q}_j}\,\mathrm{d}\bar{a}_i + m_{pi}\dot{\boldsymbol{r}}_{pi}^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{r}}_{pi}}{\partial \boldsymbol{q}_j} + \int_0^{a_i} \rho_i I_{pi}\dot{\beta}_i \frac{\partial \dot{\beta}_i}{\partial \boldsymbol{q}_j}\,\mathrm{d}\bar{a}_i + (\boldsymbol{I}_{hi}\boldsymbol{\omega}_i)^{\mathrm{T}} \frac{\partial \boldsymbol{\omega}_i}{\partial \boldsymbol{q}_j}\Bigg]. \tag{B.1b}$$

Substituting $\beta_i \equiv \bar{\boldsymbol{s}}_i^{\mathrm{T}}\boldsymbol{c}_i$ and $\dot{\beta}_i \equiv \bar{\boldsymbol{s}}_i^{\mathrm{T}}\dot{\boldsymbol{c}}_i$ from (3a), into (B.1a–b), $\frac{d}{dt}(\frac{\partial T}{\partial \dot{\boldsymbol{q}}_j})$ is obtained from (B.1a) as

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\boldsymbol{q}}_j}\right) = \sum_{i=1}^{n}\Bigg[&\int_0^{b_i} \rho_i \left\{\ddot{\boldsymbol{r}}_i^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{r}}_i}{\partial \dot{\boldsymbol{q}}_j} + \dot{\boldsymbol{r}}_i^{\mathrm{T}} \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \dot{\boldsymbol{r}}_i}{\partial \dot{\boldsymbol{q}}_j}\right)\right\}\mathrm{d}\bar{b}_i \\ &+ \int_0^{a_i} \rho_i \left\{\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}} \frac{\partial \dot{\tilde{\boldsymbol{r}}}_i}{\partial \dot{\boldsymbol{q}}_j} + \dot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}} \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \dot{\tilde{\boldsymbol{r}}}_i}{\partial \dot{\boldsymbol{q}}_j}\right)\right\}\mathrm{d}\bar{a}_i + m_{pi}\left\{\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{r}}_{pi}}{\partial \dot{\boldsymbol{q}}_j} + \dot{\boldsymbol{r}}_{pi}^{\mathrm{T}} \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \dot{\boldsymbol{r}}_{pi}}{\partial \dot{\boldsymbol{q}}_j}\right)\right\} \\ &+ \int_0^{a_i} \rho_i I_{pi}\bar{\boldsymbol{s}}_i^{\mathrm{T}}\left\{\ddot{\boldsymbol{c}}_i\bar{\boldsymbol{s}}_i^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{c}}_i}{\partial \dot{\boldsymbol{q}}_j} + \dot{\boldsymbol{c}}_i\bar{\boldsymbol{s}}_i^{\mathrm{T}} \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \dot{\boldsymbol{c}}_i}{\partial \dot{\boldsymbol{q}}_j}\right)\right\}\mathrm{d}\bar{a}_i \\ &+ \left\{\left(\boldsymbol{I}_{hi}\dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times \boldsymbol{I}_{hi}\boldsymbol{\omega}_i\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j} + (\boldsymbol{I}_{hi}\boldsymbol{\omega}_i)^{\mathrm{T}} \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j}\right)\right\}\Bigg].\end{aligned} \tag{B.2}$$

As shown in Stejskal and Valasek [47] and others, it is evident that

$$\begin{aligned}&\frac{\partial \dot{\boldsymbol{r}}_i}{\partial \dot{\boldsymbol{q}}_j} = \frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{q}_j}; \quad \frac{\partial \dot{\tilde{\boldsymbol{r}}}_i}{\partial \dot{\boldsymbol{q}}_j} = \frac{\partial \tilde{\boldsymbol{r}}_i}{\partial \boldsymbol{q}_j}; \\ &\frac{\partial \dot{\boldsymbol{r}}_{pi}}{\partial \dot{\boldsymbol{q}}_j} = \frac{\partial \boldsymbol{r}_{pi}}{\partial \boldsymbol{q}_j} \quad \text{and} \quad \frac{\partial \dot{\boldsymbol{c}}_i}{\partial \dot{\boldsymbol{q}}_j} = \frac{\partial \boldsymbol{c}_i}{\partial \boldsymbol{q}_j}.\end{aligned} \tag{B.3}$$

Hence, the 2nd, 4th, 6th, and 8th terms on the right-hand side of (B.2) are given by

$$\begin{aligned}&\frac{d}{dt}\left(\frac{\partial \dot{\boldsymbol{r}}_i}{\partial \dot{\boldsymbol{q}}_j}\right) = \frac{d}{dt}\left(\frac{\partial \boldsymbol{r}_i}{\partial \boldsymbol{q}_j}\right) = \frac{\partial \dot{\boldsymbol{r}}_i}{\partial \boldsymbol{q}_j}; \quad \frac{d}{dt}\left(\frac{\partial \dot{\tilde{\boldsymbol{r}}}_i}{\partial \dot{\boldsymbol{q}}_j}\right) = \frac{d}{dt}\left(\frac{\partial \tilde{\boldsymbol{r}}_i}{\partial \boldsymbol{q}_j}\right) = \frac{\partial \dot{\tilde{\boldsymbol{r}}}_i}{\partial \boldsymbol{q}_j}; \\ &\frac{d}{dt}\left(\frac{\partial \dot{\boldsymbol{r}}_{pi}}{\partial \dot{\boldsymbol{q}}_j}\right) = \frac{d}{dt}\left(\frac{\partial \boldsymbol{r}_{pi}}{\partial \boldsymbol{q}_j}\right) = \frac{\partial \dot{\boldsymbol{r}}_{pi}}{\partial \boldsymbol{q}_j} \quad \text{and} \quad \frac{d}{dt}\left(\frac{\partial \dot{\boldsymbol{c}}_i}{\partial \dot{\boldsymbol{q}}_j}\right) = \frac{d}{dt}\left(\frac{\partial \boldsymbol{c}_i}{\partial \boldsymbol{q}_j}\right) = \frac{\partial \dot{\boldsymbol{c}}_i}{\partial \boldsymbol{q}_j}.\end{aligned} \tag{B.4}$$

Similar to (B.4), it can also be shown that

$$\frac{d}{dt}\left(\frac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j}\right) = \frac{\partial \boldsymbol{\omega}_i}{\partial \boldsymbol{q}_j}. \tag{B.5}$$

Substituting (B.4) and (B.5) into (B.2), and using the resulting expression, along with (B.1b), the left hand side of (20), is obtained as

$$\sum_{i=1}^{n}\Bigg[\int_0^{b_i} \rho_i \ddot{\boldsymbol{r}}_i^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{r}}_i}{\partial \dot{\boldsymbol{q}}_j}\,\mathrm{d}\bar{b}_i + \int_0^{a_i} \rho_i \ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}} \frac{\partial \dot{\tilde{\boldsymbol{r}}}_i}{\partial \dot{\boldsymbol{q}}_j}\,\mathrm{d}\bar{a}_i + m_{pi}\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{r}}_{pi}}{\partial \dot{\boldsymbol{q}}_j} + \int_0^{a_i} \rho_i I_{pi}\ddot{\boldsymbol{s}}_i^{\mathrm{T}}\ddot{\boldsymbol{c}}_i\bar{\boldsymbol{s}}_i^{\mathrm{T}} \frac{\partial \dot{\boldsymbol{c}}_i}{\partial \dot{\boldsymbol{q}}_j}\,\mathrm{d}\bar{a}_i + \left(\boldsymbol{I}_{hi}\dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times \boldsymbol{I}_{hi}\boldsymbol{\omega}_i\right)\frac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j}\Bigg] = \boldsymbol{\tau}_j, \tag{B.6}$$

where cancellation of some terms happened. Next, substituting $\dot{\boldsymbol{r}}_i$, $\dot{\tilde{\boldsymbol{r}}}_i$ and $\dot{\boldsymbol{r}}_{pi}$ from (18) into (B.6), one obtains

$$\sum_{i=1}^{n}\Bigg[\int_0^{b_i}\rho_i\left\{\ddot{\boldsymbol{r}}_i^{\mathrm{T}}\frac{\partial\boldsymbol{v}_i}{\partial\dot{\boldsymbol{q}}_j}+\ddot{\boldsymbol{r}}_i^{\mathrm{T}}\frac{\partial(\boldsymbol{\omega}_i\times\bar{b}_i\boldsymbol{z}_i)}{\partial\dot{\boldsymbol{q}}_j}\right\}\mathrm{d}\bar{b}_i+\int_0^{a_i}\rho_i\left\{\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial\boldsymbol{v}_i}{\partial\dot{\boldsymbol{q}}_j}+\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial[\boldsymbol{\omega}_i\times\bar{\boldsymbol{r}}_i+\dot{\boldsymbol{u}}_i]}{\partial\dot{\boldsymbol{q}}_j}\right\}\mathrm{d}\bar{a}_i$$
$$+m_{pi}\left\{\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\frac{\partial\boldsymbol{v}_i}{\partial\dot{\boldsymbol{q}}_j}+\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\frac{\partial[\boldsymbol{\omega}_i\times\bar{\boldsymbol{r}}_{pi}+\dot{\boldsymbol{u}}_{pi}]}{\partial\dot{\boldsymbol{q}}_j}\right\}$$
$$+\int_0^{a_i}\rho_i I_{pi}\bar{\boldsymbol{s}}_i^{\mathrm{T}}\ddot{\boldsymbol{c}}_i\bar{\boldsymbol{s}}_i^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{c}}_i}{\partial\dot{\boldsymbol{q}}_j}\,\mathrm{d}\bar{a}+\left(\boldsymbol{I}_{hi}\dot{\boldsymbol{\omega}}_i+\boldsymbol{\omega}_i\times\boldsymbol{I}_{hi}\boldsymbol{\omega}_i\right)^{\mathrm{T}}\frac{\partial\boldsymbol{\omega}_i}{\partial\dot{\boldsymbol{q}}_j}\Bigg]=\boldsymbol{\tau}_j. \tag{B.7}$$

Using the vector triple product rule [24], $\boldsymbol{a}^{\mathrm{T}}(\boldsymbol{b}\times\boldsymbol{c})=(\boldsymbol{c}\times\boldsymbol{a})^{\mathrm{T}}\boldsymbol{b}$—$\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$ are any 3-dimensional Cartesian vectors, one can show that

$$\ddot{\boldsymbol{r}}_i^{\mathrm{T}}\frac{\partial(\boldsymbol{\omega}_i\times\bar{b}_i\boldsymbol{z}_i)}{\partial\dot{\boldsymbol{q}}_j}$$
$$=\bar{b}_i\ddot{\boldsymbol{r}}_i^{\mathrm{T}}\left(\frac{\partial\boldsymbol{\omega}_i}{\partial\dot{\boldsymbol{q}}_j}\times\boldsymbol{z}_i+\boldsymbol{\omega}_i\times\frac{\partial\boldsymbol{z}_i}{\partial\dot{\boldsymbol{q}}_j}\right)=\bar{b}_i(\boldsymbol{z}_i\times\ddot{\boldsymbol{r}}_i)^{\mathrm{T}}\frac{\partial\boldsymbol{\omega}_i}{\partial\dot{\boldsymbol{q}}_j}, \tag{B.8a}$$

$$\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial(\boldsymbol{\omega}_i\times\bar{\boldsymbol{r}}_i+\dot{\boldsymbol{u}}_i)}{\partial\dot{\boldsymbol{q}}_j}$$
$$=\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial(\boldsymbol{\omega}_i\times\bar{\boldsymbol{r}}_i)}{\partial\dot{\boldsymbol{q}}_j}+\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{u}}_i}{\partial\dot{\boldsymbol{q}}_j}$$
$$=\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\left(\frac{\partial\boldsymbol{\omega}_i}{\partial\dot{\boldsymbol{q}}_j}\times\bar{\boldsymbol{r}}_i+\boldsymbol{\omega}_i\times\frac{\partial\bar{\boldsymbol{r}}_i}{\partial\dot{\boldsymbol{q}}_j}\right)+\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{u}}_i}{\partial\dot{\boldsymbol{q}}_j}=\left(\bar{\boldsymbol{r}}_i\times\ddot{\tilde{\boldsymbol{r}}}_i\right)^{\mathrm{T}}\frac{\partial\boldsymbol{\omega}_i}{\partial\dot{\boldsymbol{q}}_j}+\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{u}}_i}{\partial\dot{\boldsymbol{q}}_j}, \tag{B.8b}$$

$$\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\frac{\partial(\boldsymbol{\omega}_i\times\bar{\boldsymbol{r}}_{pi}+\dot{\boldsymbol{u}}_{pi})}{\partial\dot{\boldsymbol{q}}_j}$$
$$=\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\left(\frac{\partial\boldsymbol{\omega}_i}{\partial\dot{\boldsymbol{q}}_j}\times\bar{\boldsymbol{r}}_{pi}+\boldsymbol{\omega}_i\times\frac{\partial\bar{\boldsymbol{r}}_{pi}}{\partial\dot{\boldsymbol{q}}_j}\right)+\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{u}}_{pi}}{\partial\dot{\boldsymbol{q}}_j}$$
$$=\left(\bar{\boldsymbol{r}}_{pi}\times\ddot{\boldsymbol{r}}_{pi}\right)^{\mathrm{T}}\frac{\partial\boldsymbol{\omega}_i}{\partial\dot{\boldsymbol{q}}_j}+\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{u}}_{pi}}{\partial\dot{\boldsymbol{q}}_j}. \tag{B.8c}$$

In (B.8a–c), $\partial\boldsymbol{z}_i/\partial\dot{\boldsymbol{q}}_j=\boldsymbol{O}$, $\partial\bar{\boldsymbol{r}}_i/\partial\dot{\boldsymbol{q}}_j=\boldsymbol{O}$, and $\partial\bar{\boldsymbol{r}}_{pi}/\partial\dot{\boldsymbol{q}}_j=\boldsymbol{O}$ are used as $\boldsymbol{z}_i$, $\bar{\boldsymbol{r}}_i$, and $\bar{\boldsymbol{r}}_{pi}$ are functions of $\boldsymbol{q}_j$'s only, and not $\dot{\boldsymbol{q}}_j$'s. Moreover, using (2e) and substituting $\dot{\boldsymbol{u}}_i=\boldsymbol{S}_i\dot{\boldsymbol{d}}_i$, along with $\dot{\boldsymbol{u}}_{pi}=\boldsymbol{S}_i|_{a_i}\dot{\boldsymbol{d}}_i$ in (B.8a–c), $\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{u}}_i}{\partial\dot{\boldsymbol{q}}_j}$ and $\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{u}}_{pi}}{\partial\dot{\boldsymbol{q}}_j}$ are rewritten as:

$$\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{u}}_i}{\partial\dot{\boldsymbol{q}}_j}=\ddot{\tilde{\boldsymbol{r}}}_i^{\mathrm{T}}\boldsymbol{S}_i\frac{\partial\dot{\boldsymbol{d}}_i}{\partial\dot{\boldsymbol{q}}_j}, \tag{B.9a}$$

$$\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\frac{\partial\dot{\boldsymbol{u}}_{pi}}{\partial\dot{\boldsymbol{q}}_j}=\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\boldsymbol{S}_i|_{a_i}\frac{\partial\dot{\boldsymbol{d}}_i}{\partial\dot{\boldsymbol{q}}_j}. \tag{B.9b}$$

In (B.9b), $\boldsymbol{S}_i|_{a_i}$ is the shape function of the link evaluated at its tip, i.e., $\bar{a}_i = a_i$. Substituting (B.9a–b) into (B.7) yields

$$\begin{aligned}
\sum_{i=1}^{n}\Bigg[&\int_0^{b_i}\rho_i\left\{\ddot{\boldsymbol{r}}_i^{\mathrm{T}}\frac{\partial \boldsymbol{v}_i}{\partial \dot{\boldsymbol{q}}_j}+\bar{b}_i\left(\boldsymbol{z}_i\times\ddot{\boldsymbol{r}}_i\right)^{\mathrm{T}}\frac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j}\right\}\mathrm{d}\bar{b}_i\\
&+\int_0^{a_i}\rho_i\left\{\ddot{\bar{\boldsymbol{r}}}_i^{\mathrm{T}}\frac{\partial \boldsymbol{v}_i}{\partial \dot{\boldsymbol{q}}_j}+\left(\bar{\boldsymbol{r}}_i\times\ddot{\bar{\boldsymbol{r}}}_i\right)^{\mathrm{T}}\frac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j}+\ddot{\bar{\boldsymbol{r}}}_i^{\mathrm{T}}\boldsymbol{S}_i\frac{\partial \dot{\boldsymbol{d}}_i}{\partial \dot{\boldsymbol{q}}_j}\right\}\mathrm{d}\bar{a}_i\\
&+m_{pi}\left\{\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\frac{\partial \boldsymbol{v}_i}{\partial \dot{\boldsymbol{q}}_j}+\left(\bar{\boldsymbol{r}}_{pi}\times\ddot{\boldsymbol{r}}_{pi}\right)^{\mathrm{T}}\frac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j}+\ddot{\boldsymbol{r}}_{pi}^{\mathrm{T}}\boldsymbol{S}_i|_{a_i}\frac{\partial \dot{\boldsymbol{d}}_i}{\partial \dot{\boldsymbol{q}}_j}\right\}\\
&+\int_0^{a_i}\rho_i I_{pi}\bar{\boldsymbol{s}}_i^{\mathrm{T}}\ddot{\boldsymbol{c}}_i\bar{\boldsymbol{s}}_i^{\mathrm{T}}\frac{\partial \dot{\boldsymbol{c}}_i}{\partial \dot{\boldsymbol{q}}_j}\mathrm{d}\bar{a}_i+\left(\boldsymbol{I}_{hi}\dot{\boldsymbol{\omega}}_i+\boldsymbol{\omega}_i\times\boldsymbol{I}_{hi}\boldsymbol{\omega}_i\right)^{\mathrm{T}}\frac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j}\Bigg]=\boldsymbol{\tau}_j. \qquad \text{(B.9c)}
\end{aligned}$$

Now, introduce the following definitions:

$$\frac{\partial \boldsymbol{t}_i}{\partial \dot{\boldsymbol{q}}_j}\equiv\begin{bmatrix}\dfrac{\partial \boldsymbol{v}_i}{\partial \dot{\boldsymbol{q}}_j}\\[2ex]\dfrac{\partial \boldsymbol{\omega}_i}{\partial \dot{\boldsymbol{q}}_j}\\[2ex]\dfrac{\partial \dot{\boldsymbol{d}}_i}{\partial \dot{\boldsymbol{q}}_j}\\[2ex]\dfrac{\partial \dot{\boldsymbol{c}}_i}{\partial \dot{\boldsymbol{q}}_j}\end{bmatrix};$$

$$\boldsymbol{w}_i^*\equiv\begin{bmatrix}\displaystyle\int_0^{b_i}\rho_i\ddot{\boldsymbol{r}}_i\,\mathrm{d}\bar{b}_i+\int_0^{a_i}\rho_i\ddot{\bar{\boldsymbol{r}}}_i\,\mathrm{d}\bar{a}_i+m_{pi}\ddot{\boldsymbol{r}}_{pi}\\[2ex]\displaystyle\int_0^{b_i}\rho_i\bar{b}_i\left(\boldsymbol{z}_i\times\ddot{\boldsymbol{r}}_i\right)\mathrm{d}\bar{b}_i+\int_0^{a_i}\rho_i\left(\bar{\boldsymbol{r}}_i\times\ddot{\bar{\boldsymbol{r}}}_i\right)\mathrm{d}\bar{a}_i+m_{pi}\left(\bar{\boldsymbol{r}}_{pi}\times\ddot{\boldsymbol{r}}_{pi}\right)+\left(\boldsymbol{I}_{hi}\dot{\boldsymbol{\omega}}_i+\boldsymbol{\omega}_i\times\boldsymbol{I}_{hi}\boldsymbol{\omega}_i\right)\\[2ex]\displaystyle\int_0^{a_i}\rho_i\boldsymbol{S}_i^{\mathrm{T}}\ddot{\bar{\boldsymbol{r}}}_i\,\mathrm{d}\bar{a}_i+m_{pi}\boldsymbol{S}_i|_{a_i}^{\mathrm{T}}\ddot{\boldsymbol{r}}_{pi}\\[2ex]\displaystyle\int_0^{a_i}\rho_i I_{pi}\bar{\boldsymbol{s}}_i\ddot{\boldsymbol{c}}_i^{\mathrm{T}}\bar{\boldsymbol{s}}_i\,\mathrm{d}\bar{a}_i\end{bmatrix}. \qquad \text{(B.10)}$$

Then (B.9c) is rewritten in compact form as

$$\begin{bmatrix}\left(\dfrac{\partial \boldsymbol{t}_1}{\partial \dot{\boldsymbol{q}}_j}\right)^{\mathrm{T}} & \cdots & \left(\dfrac{\partial \boldsymbol{t}_n}{\partial \dot{\boldsymbol{q}}_j}\right)^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\boldsymbol{w}_1^*\\ \vdots\\ \boldsymbol{w}_n^*\end{bmatrix}=\boldsymbol{\tau}_j \qquad \text{(B.11)}$$

which is the desired equation of motion to obtain the generalized force corresponding to the vector of the jth generalized coordinates.

References

1. Angeles, J., Lee, S.K.: The formulation of dynamical equations of holonomic mechanical systems using a natural orthogonal complement. ASME J. Appl. Mech. **55**, 243–244 (1988)
2. Ascher, U.M., Pai, D.K., Cloutier, B.P.: Forward dynamics, elimination methods, and formulation stiffness in robot simulation. Int. J. Robotics Res. **16**(6), 747–758 (1997)

3. Bauchau, O.A.: Computational scheme for flexible non-linear multibody systems. Multibody Syst. Dyn. **2**, 169–225 (1998)
4. Bauchau, O.A.: On the modeling of prismatic joints in flexible multibody systems. Trans. ASME Comput. Methods Appl. Mech. Eng. **181**, 87–105 (2000)
5. Bauchau, O.A., Wang, J.: Stability analysis of complex multibody systems. Trans. ASME J. Comput. Nonlinear Dyn. **1**, 71–80 (2006)
6. Baumgarte, J.: Stablization of constraints and integrals of motion in dynamical systems. Comput. Methods Appl. Mech. Eng. **1**, 1–16 (1972)
7. Book, W.J.: Recursive Lagrangian dynamics of flexible manipulator arms. Int. J. Robotics Res. **3**(3), 87–101 (1984)
8. Cardona, A., Geradin, M.: A beam finite element non-linear theory with finite rotations. Int. J. Numer. Methods Eng. **26**(11), 2403–2438 (2005)
9. Cetinkunt, S., Book, W.J.: Symbolic modeling of flexible manipulators. In: Proc. of IEEE Conf. on Robotics and Automation, pp. 2074–2080 (1987)
10. Chang, B., Nikravesh, P.: An adaptive constraint violation stablisation method for dynamic analysis of mechanical systems. Trans. ASME Appl. Mech. **104**, 488–492 (1985)
11. Chedmail, P., Aoustin, Y., Chevallereau, Ch.: Modeling and control of flexible robots. Int. J. Numer. Methods Eng. **32**, 1595–1619 (1991)
12. Cloutier, B.P., Pai, D.K., Ascher, U.M.: The formulation stiffness of forward dynamics algorithms and implications for robot simulation. In: Proc. of IEEE Conf. on Robotics and Automation, pp. 2816–2822. Japan, May (1995)
13. Cyril, X.: Dynamics of flexible link manipulators. Dissertation, McGill University, Canada (1988)
14. De Luca, A., Siciliano, B.: Closed-form dynamic model of planar multilink lightweight robots. IEEE Trans. Syst. Man Cybern. **21**(4), 826–838 (1991)
15. D'Eleuterio, G.M.T., Barfooy, T.D.: Just a second, we'd like to go first: a firstorder discretized formulation for structural dynamics. In: Proc. of Fourth Int. Conf. on Dynamics and Controls, pp. 1–24. London (1999)
16. Denavit, J., Hartenberg, R.S.: A kinematic notation for lower-pair mechanisms based on matrices. ASME J. Appl. Mech. **77**, 445–450 (1955)
17. Eberhard, P., Schiehlen, W.: Computational dynamics of multibody systems: history, formalisms, and applications. Trans. ASME J. Comput. Nonlinear Dyn. **1**, 3–12 (2006)
18. Ellis, R.E., Ismaeil, O.M., Carmichael, I.H.: Numerical stability of forward dynamics algorithms. In: Proc. of IEEE Conf. on Robotics and Automation, pp. 305–311. France (1992)
19. Haug, E.J.: Computer-Aided Kinematics and Dynamics of Mechanical Systems. Allyn and Bacon, Boston (1989)
20. Hwang, Y.I.: A new approach for dynamic analysis of flexible manipulator systems. Int. J. Non-Linear Mech. **40**, 925–938 (2005)
21. Ider, S.K.: Stability analysis of constraints in flexible multibody systems dynamics. Int. J. Eng. Sci. **28**(12), 1277–1290 (1990)
22. Jain, A., Rodriguez, G.: Recursive flexible multibody system dynamics using spatial operators. J. Guid. Controls Dyn. **15**, 1453–1466 (1992)
23. Jain, A., Rodriguez, G.: Sensitivity analysis for multibody systems using spatial operators. In: Int. Conf. (VI) on Methods and Models in Automation and Robotics, pp. 30–31. Poland (2000)
24. Jain, A., Rodriguez, G.: Multibody mass matrix sensitivity analysis using spatial operators. Int. J. Multiscale Comput. Eng. **1**(2–3) (2003)
25. Kamman, J.W., Huston, R.L.: Dynamics of constrained multibody systems. ASME J. Appl. Mech. **51**, 899–903 (1984)
26. Kane, T.R., Ryan, R.R., Banerjee, A.K.: Dynamics of a cantilever beam attached to a moving base. J. Guid. Control Dyn. **10**(2), 139–151 (1987)
27. Kim, S.S., Haug, E.J.: A recursive formulation for flexible multibody dynamics, Part-I: Open-loop systems. Comput. Methods Appl. Mech. Eng. **71**, 293–314 (1988)
28. Li, C.J., Shankar, T.S.: Systematic methods for efficient modeling and dynamics computation of flexible robot manipulator. IEEE Trans. Syst. Man Cybern. **23**(1), 77–94 (1993)
29. Martins, J.M., Miguel, A.B., da Costa, J.: Modeling for control of flexible robot manipulators. In: Proc. of Thematic Conf. on Multibody Dynamics. Lisbon, July 1–4, 2003
30. Meirovitch, L.: Analytical Methods in Vibrations. Macmillan, New York (1967)
31. Mohan, A., Saha, S.K.: A recursive, numerically stable, and efficient simulation algorithm for serial robots. Multibody Syst. Dyn. **17**, 291–319 (2007)
32. Neto, M.A., Ambrosio, J.: Stabilization methods for the integration of DAE in the presence of redundant constraints. Multibody Syst. Dyn. **10**(1), 81–105 (2000)
33. Nikravesh, P.E.: Computer-Aided Analysis of Mechanical Systems. Prentice Hall, New Jersey (1988)

34. Pederson, N.L.: On the formulation of flexible multibody systems with constant mass matrix. Multibody Syst. Dyn. **1**, 323–337 (1997)
35. Saha, S.K.: A decomposition of manipulator inertia matrix. IEEE Trans. Robotics Autom. **13**(2), 301–304 (1997)
36. Saha, S.K.: Dynamic modeling of serial multi-body systems using the decoupled natural orthogonal complement matrices. ASME J. Appl. Mech. **29**(2), 986–996 (1999)
37. Saha, S.K.: Analytical expression for the inverted inertia matrix of serial robots. Int. J. Robotics Res. **18**(1), 116–124 (1999)
38. Saha, S.K.: Simulation of industrial manipulators based on $\boldsymbol{UDU}^{\mathrm{T}}$ decomposition of inertia matrix. Multibody Syst. Dyn. **9**, 63–85 (2003)
39. Saha, S.K., Angeles, J.: Dynamics of nonholonomic mechanical systems using a natural orthogonal complement. ASME J. Appl. Mech. **58**, 238–243 (1991)
40. Schiehlen, W.O.: Recent developments in multibody dynamics. J. Mech. Sci. Technol. **19**(1), 129–141 (2005)
41. Shabana, A.A.: Dynamics of flexible bodies using generalized Newton–Euler equation. ASME J. Dyn. Syst. Meas. Control **112**(3), 496–503 (1990)
42. Shabana, A.: Flexible multibody dynamics: review of past and recent developments. Multibody Syst. Dyn. **1**, 189–222 (1997)
43. Shabana, A.A.: Dynamics of Multibody Systems. Cambridge University Press, Cambridge (2005)
44. Sharf, I.: Nonlinear strain measure, shape functions and beam elements for dynamics of flexible beams. Multibody Syst. Dyn. **3**, 189–205 (1999)
45. Sharf, I., Damaren, C.: Simulation of flexible link manipulators: basis functions and non-linear terms in motion equations. In: Proc. of IEEE Conf. on Robotics and Automation, pp. 1956–1962. France (1992)
46. Shim, Y.J., Sung, Y.: Stability and four posture control for non-holonomic mobile robots. IEEE Trans. Robotics Autom. **20**(1), 148–154 (2004)
47. Stejskal, V., Valasek, M.: Kinematics and Dynamics of Machinery. M. Dekkar, New York (1996)
48. Strang, G.: Linear Algebra and Its Applications. H.B. Jovanovich Pub., Florida (1980)
49. Theodore, R.J., Ghosal, A.: Comparison of the assumed modes and finite elements modes for flexible multilink manipulators. Int. J. Robotics Res. **14**(2), 91–111 (1995)
50. Thompson, W.T.: Theory of Vibration with Applications. Prentice Hall, London (1988)
51. Usoro, P.B., Nadira, R., Mahil, S.S.: A finite element/Lagrangian approach to modeling light weight flexible manipulators. ASME J. Dyn. Syst. Meas. Control **108**, 198–205 (1986)
52. Walker, M.W., Orin, D.E.: Efficient dynamic computer simulation of robotic mechanisms. ASME J. Dyn. Syst. Meas. Control **104**, 205–211 (1982)
53. Wasfy, T.M., Noor, A.K.: Computational strategies for flexible multibody systems. ASME J. Appl. Mech. Rev. **56**(6), 553–613 (2003)
54. Wehage, R.A., Haug, E.J.: Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems. ASME J. Mech. Des. **104**, 247–255 (1982)