

Screw and Lie group theory in multibody dynamics

Recursive algorithms and equations of motion of tree-topology systems

Andreas Müller¹

Received: 12 November 2016 / Accepted: 16 June 2017 / Published online: 19 July 2017
© The Author(s) 2017. This article is published with open access at Springerlink.com

Abstract Screw and Lie group theory allows for user-friendly modeling of multibody systems (MBS), and at the same they give rise to computationally efficient recursive algorithms. The inherent frame invariance of such formulations allows to use arbitrary reference frames within the kinematics modeling (rather than obeying modeling conventions such as the Denavit–Hartenberg convention) and to avoid introduction of joint frames. The computational efficiency is owed to a representation of twists, accelerations, and wrenches that minimizes the computational effort. This can be directly carried over to dynamics formulations. In this paper, recursive $O(n)$ Newton–Euler algorithms are derived for the four most frequently used representations of twists, and their specific features are discussed. These formulations are related to the corresponding algorithms that were presented in the literature. Two forms of MBS motion equations are derived in closed form using the Lie group formulation: the so-called Euler–Jourdain or “projection” equations, of which Kane’s equations are a special case, and the Lagrange equations. The recursive kinematics formulations are readily extended to higher orders in order to compute derivatives of the motions equations. To this end, recursive formulations for the acceleration and jerk are derived. It is briefly discussed how this can be employed for derivation of the linearized motion equations and their time derivatives. The geometric modeling allows for direct application of Lie group integration methods, which is briefly discussed.

Keywords Multibody system dynamics · Relative coordinates · Recursive algorithms · $O(n)$ · Screws · Lie groups · Newton–Euler equations · Lagrange equations · Kane’s equations · Euler–Jourdain equations · Projection equations · Lie group integration · Linearization

Dedicated to Prof. Peter Maißer

✉ A. Müller
a.mueller@jku.at

¹ Institute of Robotics, Johannes Kepler University, Altenberger Str. 69, 4040 Linz, Austria

1 Introduction

The core task in computational multibody system (MBS) dynamics is to either construct the equations of motion (EOM) explicitly that can be written for an unconstrained tree-topology MBS in the form

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\dot{\mathbf{q}}, \mathbf{q})\dot{\mathbf{q}} = \mathbf{Q}(\dot{\mathbf{q}}, \mathbf{q}, t) \quad (1)$$

in a way that is easy to pursue, or to evaluate them for given $(\ddot{\mathbf{q}}, \dot{\mathbf{q}}, \mathbf{q})$ and t , respectively to solve them, in a computationally efficient way for $\mathbf{q}(t)$. In continuation of [60], the aim of this paper is to present established $O(n)$ formulations in a common geometric setting and to show that this setting allows for a flexible and user-friendly MBS modeling.

Screw and Lie group theory provides a geometric framework that allows for achieving optimal computational performance and at the same time allows for an intuitive and flexible modeling. In particular, it gives rise to a formulation of the MBS kinematics that does not involve body-fixed joint frames. The kinematics modeling is indeed reflected in the formulation used to evaluate the EOM. A central concept is the representation of velocities (twists) as screws. Four different variants were recalled in [60]. In this paper, their application to dynamics modeling is reviewed. A well-known approach, which exploits the fact that rigid-body twists are screws, is the so-called “spatial vector” formulation introduced in [26, 29], respectively the so-called “spatial operator algebra”, which was formalized in [73]. The latter is the basis for the $O(n)$ forward dynamics algorithms introduced in [30, 37, 38, 44, 72, 74]. The fundamental operation underlying these formulations is the frame transformations of screws, that is, twists and wrenches. The fact that the latter can be expressed in terms of compact matrix operations gave rise to a matrix formulation for the MBS kinematic and dynamics [5, 42, 43, 83] using screw algebra. While these formulations make merely use of the algebraic properties of screws (e.g. velocities, accelerations, wrenches), several algorithms for generating the EOM of MBS with tree topology were reported, which also exploit the fact that finite rigid body motions constitute the Lie group $SE(3)$ whose Lie algebra $se(3)$ is isomorphic to the algebra of screws [15, 23, 24, 32, 33]. The central relation is the *product of exponentials* (POE) introduced in [15]. The important feature of such a geometric Lie group formulation is the frame invariance, which makes it independent from any modeling convention like the Denavit–Hartenberg convention. This allows for direct processing of CAD data and gives further rise to numerically advantageous Lie group time integration methods. Yet there is no established Lie group algorithm for the generation respectively evaluation of the EOM that takes full advantage of the freedom to choose different motion representations enabled by the frame invariance.

This paper is organized as follows. Recursive relations for the acceleration and jerk, and thus for the time derivatives of the Jacobians, are first derived in Sect. 2. The Newton–Euler equations for the four different representations of twists introduced in [60] are then recalled in Sect. 3. The corresponding recursive $O(n)$ inverse dynamics algorithm for evaluating the EOM are presented in Sect. 4. The body-fixed algorithm is similar to that in [2, 7, 30, 34, 35, 44, 45, 67, 68, 70, 71, 76], the hybrid formulation to that in [1, 6, 37, 38, 73, 74], and the spatial formulation to that in [29]. Two versions of the EOM in closed form are presented in Sect. 5. In Sect. 5.1, the Euler–Jourdain respectively “projection” equations [14, 84] are presented that, together with the screw formulation of MBS kinematics, allow for an efficient MBS modeling in terms of readily available geometric data. In Sect. 5.2, a closed form of the Lagrangian EOM is presented using the Lie group approach. It should be noticed that the presented formulations allow for modeling MBS without introduction of joint frames while applying the recursive kinematics and dynamics algorithm that is deemed best suited.

The significance of the Lie group formulation for the linearization of the EOM and the determination of derivative of the EOM with respect to geometric design parameters and time derivatives is discussed in Sect. 6. Finally, in Sect. 7 the application of Lie group integration methods is briefly discussed. The kinematic relations presented in [60] are summarized in Appendix A. The basic Lie group background can be found in [47, 63, 75].

2 Acceleration, jerk, and partial derivatives of Jacobians

Besides the compact description of finite and instantaneous motions of a system of articulated bodies, a prominent feature of the screw theoretical approach is that it allows for expressing the partial derivatives explicitly in terms geometric objects. Moreover, the analytic formulation of the kinematics using the POE gives rise to compact expressions for higher derivatives of the instantaneous joint screws, that is, of the Jacobian, which may be relevant for sensitivity analysis and linearization of motion equations. In this section, results for the acceleration and jerk of a kinematic chain are presented for the body-fixed, spatial, and hybrid representation. The corresponding relations for the mixed representation are readily found from either one of these using the relations in Table 3 of [60].

2.1 Body fixed representation

Starting from (102), the body-fixed acceleration is $\dot{\mathbf{V}}_i^b = \mathbf{J}_i^b \ddot{\mathbf{q}} + \dot{\mathbf{J}}_i^b \dot{\mathbf{q}}$ or, explicitly in terms of the body-fixed instantaneous screw coordinates,

$$\dot{\mathbf{V}}_i^b = \sum_{j \leq i} \mathbf{J}_{i,j}^b \ddot{q}_j + \sum_{j \leq i} \sum_{k \leq i} \frac{\partial}{\partial q_k} \mathbf{J}_{i,j}^b \dot{q}_j \dot{q}_k. \tag{2}$$

Using the matrix form of (104), the partial derivatives of the instantaneous screw coordinates are

$$\frac{\partial}{\partial q_k} \widehat{\mathbf{J}}_{i,j}^b = \frac{\partial}{\partial q_k} (\mathbf{C}_i^{-1} \mathbf{C}_j) \mathbf{A}_j^{-1} \widehat{\mathbf{Y}}_j \mathbf{A}_j \mathbf{C}_i^{-1} \mathbf{C}_j + \mathbf{C}_i^{-1} \mathbf{C}_j \mathbf{A}_j^{-1} \widehat{\mathbf{Y}}_j \mathbf{A}_j \frac{\partial}{\partial q_k} (\mathbf{C}_j^{-1} \mathbf{C}_i). \tag{3}$$

This can be evaluated with help of the POE formula (94) as

$$\begin{aligned} & \frac{\partial}{\partial q_k} (\mathbf{C}_i^{-1} \mathbf{C}_j) \\ &= \frac{\partial}{\partial q_k} (\mathbf{A}_i^{-1} \exp(-\widehat{\mathbf{Y}}_i q_i) \cdots \exp(-\widehat{\mathbf{Y}}_{j+1} q_{j+1}) \mathbf{A}_j) \\ &= -\mathbf{A}_i^{-1} \exp(-\widehat{\mathbf{Y}}_i q_i) \cdots \exp(-\widehat{\mathbf{Y}}_{k+1} q_{k+1}) \widehat{\mathbf{Y}}_k \exp(-\widehat{\mathbf{Y}}_k q_k) \cdots \exp(-\widehat{\mathbf{Y}}_{j+1} q_{j+1}) \mathbf{A}_j \\ &= -\mathbf{C}_i^{-1} \mathbf{C}_k \mathbf{A}_k^{-1} \widehat{\mathbf{Y}}_k \mathbf{A}_k \mathbf{C}_k^{-1} \mathbf{C}_j = -\mathbf{C}_i^{-1} \mathbf{C}_k \mathbf{A}_k^{-1} \widehat{\mathbf{Y}}_k \mathbf{A}_k \mathbf{C}_k^{-1} \mathbf{C}_i \mathbf{C}_i^{-1} \mathbf{C}_j \\ &= -\widehat{\mathbf{J}}_{i,k}^b \mathbf{C}_i^{-1} \mathbf{C}_j, \quad j \leq k \leq i, \end{aligned} \tag{4}$$

and in the same way, it follows that

$$\frac{\partial}{\partial q_k} (\mathbf{C}_j^{-1} \mathbf{C}_i) = \mathbf{C}_j^{-1} \mathbf{C}_i \widehat{\mathbf{J}}_{i,j}^b, \quad j \leq k \leq i. \tag{5}$$

Insertion into (3) yields $\frac{\partial}{\partial q_k} \widehat{\mathbf{J}}_{i,j}^b = \widehat{\mathbf{J}}_{i,j}^b \widehat{\mathbf{J}}_{i,k}^b - \widehat{\mathbf{J}}_{i,k}^b \widehat{\mathbf{J}}_{i,j}^b$, and noting (114), the final expression is

$$\frac{\partial \mathbf{J}_{i,j}^b}{\partial q_k} = [\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b], \quad j < k \leq i. \tag{6}$$

Hence the partial derivative of the instantaneous joint screw $\mathbf{J}_{i,j}^b$ with respect to q_k is simply the screw product (115) of $\mathbf{J}_{i,j}^b$ and $\mathbf{J}_{i,k}^b$. The final expression for the acceleration attains a very compact form

$$\dot{\mathbf{V}}_i^b = \sum_{j \leq i} \mathbf{J}_{i,j}^b \ddot{q}_j + \sum_{j < k \leq i} [\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b] \dot{q}_j \dot{q}_k. \tag{7}$$

Indeed, the same result would be obtained using (104) in terms of \mathbf{Y}_i . This expression has been derived, using different notations, for instance, in [15, 50, 63, 67].

Equation (7) can be summarized for all bodies $i = 1, \dots, n$ using the system twist (112) and system Jacobian (113). To this end, the derivative (6) is rewritten as

$$\frac{\partial \mathbf{J}_{i,j}^b}{\partial q_k} = [\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b] = \mathbf{Ad}_{\mathbf{C}_{i,k}} [\mathbf{J}_{i,j}^b, {}^k \mathbf{X}_k] = -\mathbf{Ad}_{\mathbf{C}_{i,k}} \mathbf{ad}_{\mathbf{X}_k} \mathbf{J}_{i,j}^b, \quad j < k \leq i, \tag{8}$$

so that

$$\dot{\mathbf{J}}_{i,j}^b = \sum_{j < k \leq i} [\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b] \dot{q}_k = - \sum_{j < k \leq i} \mathbf{Ad}_{\mathbf{C}_{i,k}} \mathbf{ad}_{\mathbf{X}_k} \mathbf{J}_{i,j}^b \dot{q}_k.$$

Noticing that $\mathbf{ad}_{\mathbf{X}_k} \mathbf{J}_{k,k}^b = \mathbf{0}$, the time derivative of the body-fixed system Jacobian factors as

$$\dot{\mathbf{J}}^b(\mathbf{q}, \dot{\mathbf{q}}) = -\mathbf{A}^b(\mathbf{q}) \mathbf{a}^b(\dot{\mathbf{q}}) \mathbf{A}^b(\mathbf{q}) \mathbf{X}^b = -\mathbf{A}^b(\mathbf{q}) \mathbf{a}^b(\dot{\mathbf{q}}) \mathbf{J}^b(\mathbf{q}) \tag{9}$$

with \mathbf{A}^b defined in (24) of [60] and with

$$\mathbf{a}^b(\dot{\mathbf{q}}) := \text{diag}(\dot{q}_1 \mathbf{ad}_{\mathbf{X}_1}, \dots, \dot{q}_n \mathbf{ad}_{\mathbf{X}_n}). \tag{10}$$

Hence the system acceleration is given in compact matrix form as

$$\dot{\mathbf{V}}^b = \mathbf{J}^b \ddot{\mathbf{q}} - \mathbf{A}^b \mathbf{a}^b \mathbf{J}^b \dot{\mathbf{q}} = \mathbf{J}^b \ddot{\mathbf{q}} - \mathbf{A}^b \mathbf{a}^b \mathbf{V}^b. \tag{11}$$

Remark 1 (Overall inverse kinematics solution) Relation (11) gives rise to a solution of the inverse kinematics problem on acceleration level, that is, the generalized accelerations for given configurations, twists, and accelerations of the bodies. The unique solution is

$$\ddot{\mathbf{q}} = ((\mathbf{X}^b)^T \mathbf{X}^b)^{-1} (\mathbf{X}^b)^T ((\mathbf{I} - \mathbf{D}^b) \dot{\mathbf{V}}^b + \mathbf{a}^b \mathbf{V}^b), \tag{12}$$

which is indeed the time derivative of (26) in [60]. In components, this gives the acceleration of the individual joints as $\ddot{q}_i = {}^i \mathbf{X}_i^T (\dot{\mathbf{V}}_i^b - \mathbf{Ad}_{\mathbf{C}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^b + \dot{q}_i [{}^i \mathbf{X}_i, \mathbf{V}_i^b]) / \|{}^i \mathbf{X}_i\|^2$.

A further time derivative of the twist yields the jerk of a body, which requires a further partial derivative of the Jacobian. Starting from (6) and using the Jacobi identity (117) and the bilinearity $\frac{\partial}{\partial q_k} [\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b] = [\frac{\partial}{\partial q_k} \mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b] + [\mathbf{J}_{i,j}^b, \frac{\partial}{\partial q_k} \mathbf{J}_{i,k}^b]$, the nonzero second partial derivative is found as

$$\frac{\partial^2 \mathbf{J}_{i,j}^b}{\partial q_k \partial q_r} = \begin{cases} [[\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b], \mathbf{J}_{i,r}^b], & j < k \leq r \leq i, \\ [[\mathbf{J}_{i,j}^b, \mathbf{J}_{i,r}^b], \mathbf{J}_{i,k}^b], & j < r < k \leq i. \end{cases} \tag{13}$$

This gives rise to an explicit form for the body-fixed jerk:

$$\begin{aligned} \ddot{\mathbf{V}}_i^b = & \sum_{j \leq i} \mathbf{J}_{i,j}^b \ddot{q}_j + 2 \sum_{j < k \leq i} [\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b] \dot{q}_j \dot{q}_k \\ & + \sum_{j < k \leq i} [\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b] \dot{q}_j \ddot{q}_k + 2 \sum_{j < k \leq r \leq i} [[\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b], \mathbf{J}_{i,r}^b] \dot{q}_j \dot{q}_k \dot{q}_r. \end{aligned} \tag{14}$$

Thus only computationally simple nested screw products are required to compute the terms that are quadratic and cubic in \dot{q}_j, \dot{q}_k . The same applies to higher derivatives (which, for instance, are required for motion planning). The explicit form of the ν th-order partial derivative was presented in [55]:

$$\frac{\partial^\nu \mathbf{J}_{i,j}^b}{\partial q_{\alpha_1} \partial q_{\alpha_2} \cdots \partial q_{\alpha_\nu}} = [\dots [[[\mathbf{J}_{i,j}^b, \mathbf{J}_{i,\beta_1}^b], \mathbf{J}_{i,\beta_2}^b], \mathbf{J}_{i,\beta_3}^b] \dots, \mathbf{J}_{i,\beta_\nu}^b], \quad j < \beta_1 \leq \beta_2 \leq \dots \leq \beta_\nu \leq i, \tag{15}$$

where $\beta_1 \leq \beta_2 \leq \dots \leq \beta_\nu$ is the ordered sequence of the indices $\alpha_1, \dots, \alpha_\nu$. Clearly, the closed-form expressions become very involved. Their explicit determination can be avoided by recursive evaluation [55].

2.2 Spatial representation

Proceeding in the same way as for (3), the partial derivative of the spatial Jacobian is obtained as

$$\boxed{\frac{\partial \mathbf{J}_j^s}{\partial q_k} = [\mathbf{J}_k^s, \mathbf{J}_j^s], \quad k < j.} \tag{16}$$

Since the spatial representation \mathbf{J}_j^s is intrinsic to the joint j , rather than related to a body as is (104), the time derivative can be expressed as

$$\dot{\mathbf{J}}_j^s = \sum_{k \leq j} [\mathbf{J}_k^s, \mathbf{J}_j^s] \dot{q}^k = \left[\sum_{k \leq j} \mathbf{J}_k^s, \mathbf{J}_j^s \right] \dot{q}_k = [\mathbf{V}_j^s, \mathbf{J}_j^s]. \tag{17}$$

This relation reconfirms the special properties of spatial twists that are advantageous for recursive implementations. It may be considered as an extension of Euler’s formula for the time derivative of vectors resolved in moving frames to screws. For this reason, Featherstone [27, 29] termed the Lie bracket the “spatial cross product”. The spatial acceleration is therewith

$$\dot{\mathbf{V}}_i^s = \sum_{j \leq i} \mathbf{J}_j^s \ddot{q}_j + \sum_{k < j \leq i} [\mathbf{J}_k^s, \mathbf{J}_j^s] \dot{q}_j \dot{q}_k = \sum_{j \leq i} (\mathbf{J}_j^s \ddot{q}_j + [\mathbf{V}_j^s, \mathbf{J}_j^s] \dot{q}_j). \tag{18}$$

In matrix form the overall spatial acceleration can be summarized as

$$\dot{\mathbf{V}}^s = \mathbf{J}^s \ddot{\mathbf{q}} + \mathbf{Lb}^s \mathbf{J}^s \dot{\mathbf{q}} = \mathbf{J}^s \ddot{\mathbf{q}} + \mathbf{Lb}^s \mathbf{V}^s \tag{19}$$

with

$$\mathbf{b}^s(\mathbf{V}^s) := \text{diag}(\mathbf{ad}_{\mathbf{v}_1^s}, \dots, \mathbf{ad}_{\mathbf{v}_n^s}) \tag{20}$$

and \mathbf{L} being the lower triangular block identity matrix. A solution for $\ddot{\mathbf{q}}$ similar to (11) exists.

The second partial derivative of the spatial Jacobian is

$$\frac{\partial^2 \mathbf{J}_i^s}{\partial q_k \partial q_j} = \begin{cases} [\mathbf{J}_k^s, [\mathbf{J}_j^s, \mathbf{J}_i^s]], & k < j < i, \\ [\mathbf{J}_j^s, [\mathbf{J}_k^s, \mathbf{J}_i^s]], & j \leq k < i. \end{cases} \tag{21}$$

Therewith the spatial representation of the jerk of body i is found as

$$\ddot{\mathbf{v}}_i^s = \sum_{j \leq i} \left(\mathbf{J}_j^s \ddot{q}_j + 2[\mathbf{V}_j^s, \mathbf{J}_j^s] \dot{q}_j + \sum_{k \leq j} [\mathbf{J}_k^s \ddot{q}_k, \mathbf{J}_j^s] \dot{q}^j + [\mathbf{V}_{j-1}^s + \mathbf{V}_j^s - \mathbf{V}_i^s, [\mathbf{V}_j^s, \mathbf{J}_j^s]] \dot{q}_j \right) \tag{22}$$

$$\begin{aligned} &= \sum_{j \leq i} \left(\mathbf{J}_j^s \ddot{q}_j + [[\mathbf{V}_j^s, \mathbf{J}_j^s], \mathbf{V}_i^s - 2\mathbf{V}_j^s] \dot{q}_j + \left[\sum_{k \leq j} \mathbf{J}_k^s \ddot{q}_k + [\mathbf{V}_j^s, \mathbf{J}_j^s] \dot{q}_j, \mathbf{J}_j^s \right] \dot{q}_j \right. \\ &\quad \left. + 2[\mathbf{V}_j^s, \mathbf{J}_j^s] \dot{q}_j \right). \end{aligned} \tag{23}$$

The instantaneous joint screws (105) and thus their derivatives (16) and (21) are independent of a particular body. The closed form of the ν th-order partial derivative has been reported in [54]:

$$\begin{aligned} \frac{\partial^\nu \mathbf{J}_i^s}{\partial q_{\alpha_1} \partial q_{\alpha_2} \cdots \partial q_{\alpha_\nu}} &= [\mathbf{J}_{\beta_\nu}^s, [\mathbf{J}_{\beta_{\nu-1}}^s, [\mathbf{J}_{\beta_{\nu-2}}^s, \cdots [\mathbf{J}_{\beta_1}^s, \mathbf{J}_i^s] \cdots]]], \quad \beta_\nu \leq \beta_{\nu-1} \leq \cdots \leq \beta_1 < i, \\ &= \mathbf{ad}_{\mathbf{J}_{\beta_\nu}^s} \mathbf{ad}_{\mathbf{J}_{\beta_{\nu-1}}^s} \mathbf{ad}_{\mathbf{J}_{\beta_{\nu-2}}^s} \cdots \mathbf{ad}_{\mathbf{J}_{\beta_1}^s} \mathbf{J}_i^s, \quad \beta_\nu \leq \beta_{\nu-1} \leq \cdots \leq \beta_1 < i, \\ &= \left[\mathbf{J}_{\beta_\nu}^s, \frac{\partial^{\nu-1} \mathbf{J}_i^s}{\partial q_{\beta_1} \partial q_{\beta_2} \cdots \partial q_{\beta_{\nu-1}}} \right], \quad \beta_\nu \leq \beta_{\nu-1} < i, \end{aligned} \tag{24}$$

where again $\beta_\nu \leq \beta_{\nu-1} \leq \cdots \leq \beta_1$ is the ordered sequence of the indices $\alpha_1, \dots, \alpha_\nu$. The last form in (24) allows for a recursive determination. Moreover, a recursive formulation for the time derivative of spatial twists has been reported in [56]. Together with the very concise form (17), this makes the spatial representation computationally very attractive.

2.3 Hybrid form

The results in Sect. 2.1 can be carried over to the hybrid twist using relation (107). As in (119), denote by $\mathbf{J}_{i,k}^v$ and $\mathbf{J}_{i,k}^h$ the screw coordinate vectors comprising respectively the linear and angular parts of the column of the hybrid Jacobian, so that $\mathbf{J}_{i,k}^h = \overset{\omega}{\mathbf{J}}_{i,k}^h + \overset{v}{\mathbf{J}}_{i,k}^h$. Then

$$\begin{aligned} \frac{\partial \mathbf{J}_{i,j}^h}{\partial q_k} &= \frac{\partial \mathbf{Ad}_{\mathbf{R}_i}}{\partial q_k} \mathbf{J}_{i,j}^b + \mathbf{Ad}_{\mathbf{R}_i} \frac{\partial \mathbf{J}_{i,j}^b}{\partial q_k} = \mathbf{ad}_{\overset{\omega}{\mathbf{J}}_{i,k}^h} \mathbf{Ad}_{\mathbf{R}_i} \mathbf{J}_{i,j}^b + \mathbf{Ad}_{\mathbf{R}_i} [\mathbf{J}_{i,j}^b, \mathbf{J}_{i,k}^b] \\ &= [\overset{\omega}{\mathbf{J}}_{i,k}^h, \mathbf{J}_{i,j}^h] + [\mathbf{J}_{i,j}^h, \mathbf{J}_{i,k}^h] = [\mathbf{J}_{i,j}^h, \mathbf{J}_{i,k}^h - \overset{\omega}{\mathbf{J}}_{i,k}^h], \end{aligned} \tag{25}$$

and thus

$$\boxed{\frac{\partial \mathbf{J}_{i,j}^h}{\partial q_k} = [\mathbf{J}_{i,j}^h, \overset{v}{\mathbf{J}}_{i,k}^h] = -\mathbf{ad}_{\overset{v}{\mathbf{J}}_{i,k}^h} \mathbf{J}_{i,j}^h, \quad j \leq k \leq i.} \tag{26}$$

The similarity to (6) is apparent. The difference is that the convective term due to the angular motion is missing, which is why only \mathbf{J} appears. The time derivative of the hybrid Jacobian can thus be expressed as

$$\dot{\mathbf{J}}_{i,j}^h = \sum_{k \leq j} [\mathbf{J}_{i,j}^h, \mathbf{J}_{i,k}^v] \dot{q}_k = [\mathbf{J}_{i,j}^h, \Delta \mathbf{V}_{j-1,i}^v], \tag{27}$$

where $\Delta \mathbf{V}_{j-1,i}^h := \mathbf{V}_i^h - \mathbf{Ad}_{\mathbf{r}_{i,j-1}} \mathbf{V}_{j-1}^h$ is the relative hybrid twist of body $j - 1$ and i as observed in the BFR on body i . A simpler relation is obtained by directly differentiating (106):

$$\begin{aligned} \dot{\mathbf{J}}_{i,j}^h &= (\mathbf{ad}_{\mathbf{r}_{i,j}} + \mathbf{Ad}_{\mathbf{r}_{i,j-1}} \mathbf{ad}_{\omega_j^s})^0 \mathbf{X}_j^j \\ &= \mathbf{Ad}_{\mathbf{r}_{i,j-1}} (\mathbf{ad}_{\mathbf{d}_{i,j}} + \mathbf{ad}_{\omega_j^s})^0 \mathbf{X}_j^j = \mathbf{Ad}_{\mathbf{r}_{i,j-1}} (\mathbf{ad}_{\mathbf{v}_j^s} - \mathbf{ad}_{\mathbf{r}_i})^0 \mathbf{X}_j^j. \end{aligned} \tag{28}$$

This yields the following explicit expressions for the hybrid acceleration:

$$\dot{\mathbf{V}}_i^h = \sum_{j \leq i} \mathbf{J}_j^h \ddot{q}_j + \sum_{j \leq k \leq i} [\mathbf{J}_{i,j}^h, \mathbf{J}_{i,k}^v] \dot{q}_j \dot{q}_k = \sum_{j \leq i} (\mathbf{J}_j^h \ddot{q}_j + [\mathbf{J}_{i,j}^h, \Delta \mathbf{V}_{j-1,i}^v] \dot{q}_j) \tag{29}$$

$$= \sum_{j \leq i} (\mathbf{J}_j^h \ddot{q}_j + (\mathbf{ad}_{\mathbf{r}_{i,j}} + \mathbf{Ad}_{\mathbf{r}_{i,j}} \mathbf{ad}_{\omega_j^s})^0 \mathbf{X}_j^j \dot{q}_j). \tag{30}$$

For the second derivative, it is simplest to start from (28), and a straightforward calculation yields

$$\ddot{\mathbf{J}}_{i,j}^h = (\mathbf{ad}_{\mathbf{r}_{i,j}} + 2\mathbf{ad}_{\mathbf{r}_{i,j}} \mathbf{ad}_{\omega_j^s} + \mathbf{Ad}_{\mathbf{r}_{i,j}} (\mathbf{ad}_{\omega_j^s} + \mathbf{ad}_{\omega_j^s} \mathbf{ad}_{\omega_j^s}))^0 \mathbf{X}_j^j. \tag{31}$$

The jerk in hybrid representation can thus be written as

$$\ddot{\mathbf{V}}_i^h = \sum_{j \leq i} (\mathbf{J}_j^h \ddot{\ddot{q}}_j + 2\mathbf{ad}_{\mathbf{r}_{i,j}} \ddot{q}_j + (\mathbf{ad}_{\mathbf{r}_{i,j}} + 2\mathbf{ad}_{\mathbf{r}_{i,j}} \mathbf{ad}_{\omega_j^s}) \dot{q}_j) \tag{32}$$

$$+ \mathbf{Ad}_{\mathbf{r}_{i,j}} (2\mathbf{ad}_{\omega_j^s} \ddot{q}_j + (\mathbf{ad}_{\omega_j^s} + \mathbf{ad}_{\omega_j^s} \mathbf{ad}_{\omega_j^s}) \dot{q}_j)^0 \mathbf{X}_j^j. \tag{33}$$

These are the core relations in the so-called ‘‘spatial vector’’ formulation (i.e. using the hybrid representation of twists) [30, 37, 38, 44, 72, 74]. In this context the Lie bracket, respectively screw product (115) has been termed the ‘‘spatial cross product’’ [27, 29].

2.4 Mixed representation

With (101), employing the results for the mixed representation yields

$$\mathbf{j}_{ij}^m = \begin{pmatrix} \mathbf{R}_i^T & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \mathbf{j}_{ij}^h, \quad \dot{\mathbf{v}}_i^m = \begin{pmatrix} \mathbf{R}_i^T & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \dot{\mathbf{v}}_i^h, \quad \ddot{\mathbf{v}}_i^m = \begin{pmatrix} \mathbf{R}_i^T & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \ddot{\mathbf{v}}_i^h. \tag{34}$$

3 Newton–Euler equations in various representations

3.1 Spatial representation

Consider a rigid body with body-fixed BFR $\mathcal{F}_b = \{\Omega; \vec{e}_{b,1}, \vec{e}_{b,2}, \vec{e}_{b,3}\}$ located at an arbitrary point Ω . Denote the inertia matrix with respect to this BFR by \mathbf{M}^b ; see (41). The configu-

ration of the BFR \mathcal{F}_b is described by $\mathbf{C} = (\mathbf{R}, \mathbf{r})$. The spatial inertia matrix expressed in the IFR is then

$$\mathbf{M}^s = \mathbf{Ad}_C^{-T} \mathbf{M}^b \mathbf{Ad}_C^{-1}. \tag{35}$$

The spatial canonical momentum coscrew $\mathbf{\Pi}^s = (\mathbf{L}^s, \mathbf{P}^s)^T \in se^*(3)$, conjugate to the spatial twist, is thus

$$\mathbf{\Pi}^s = \mathbf{M}^s \mathbf{V}^s = \mathbf{Ad}_C^{-T} \mathbf{M}^b \mathbf{Ad}_C^{-1} \mathbf{V}^s = \mathbf{Ad}_C^{-T} \mathbf{\Pi}^b. \tag{36}$$

The momentum balance yields the Newton–Euler (NE) equations in spatial representation, which attains the simple form

$$\dot{\mathbf{\Pi}}^s = \mathbf{W}^s, \tag{37}$$

where $\mathbf{W}^s = (\mathbf{t}^s, \mathbf{f}^s)^T$ is the applied wrench with spatial torque $\mathbf{t}^s \equiv {}^0\mathbf{t}^0$ and force $\mathbf{f}^s \equiv {}^0\mathbf{f}$, both measured and resolved in the IFR. The momentum balance equation (37) is the simplest form possible, which is achieved by using the spatial representation of twist, wrench, and momentum. Firstly, it does not involve any vectorial operation, for example, cross products. Secondly, it is also numerically advantageous: any numerical discretization of the ODE (37) easily preserves the spatial momentum in the absence of external wrenches. This has been discussed already by Borri [12]. In this context the spatial formulation is called the fixed pole equation. In a recent paper [31] the advantages of this form are exploited for geometrically exact modeling of beams.

An explicit and compact form in terms of the spatial twist is found, introducing (36) and using

$$\dot{\mathbf{M}}^s = -\mathbf{ad}_{\mathbf{V}^s}^T \mathbf{M}^s - \mathbf{M}^s \mathbf{ad}_{\mathbf{V}^s} \tag{38}$$

along with $\mathbf{ad}_{\mathbf{V}^s} \mathbf{V}^s = \mathbf{0}$, as

$$\boxed{\mathbf{W}^s = \mathbf{M}^s \dot{\mathbf{V}}^s - \mathbf{ad}_{\mathbf{V}^s}^T \mathbf{M}^s \mathbf{V}^s}. \tag{39}$$

Remark 2 Writing (39) as $\mathbf{W}^s = \mathbf{M}^s \dot{\mathbf{V}}^s + \mathbf{C}^s \mathbf{V}^s$ (with $\mathbf{C}^s := -\mathbf{ad}_{\mathbf{V}^s}^T \mathbf{M}^s$) shows that $\dot{\mathbf{M}}^s - 2\mathbf{C}^s = \mathbf{ad}_{\mathbf{V}^s}^T \mathbf{M}^s - \mathbf{M}^s \mathbf{ad}_{\mathbf{V}^s}$ is skew symmetric. This property is called the skew symmetry of the motion equations [63].

3.2 Body-fixed representation

Let $\mathcal{F}_c = \{C; \vec{e}_{c,1}, \vec{e}_{c,2}, \vec{e}_{c,3}\}$ be a body-fixed frame located at the COM. Its configuration is described by $C_c = (\mathbf{R}_c, \mathbf{r}_c)$. The body-fixed twist of the COM frame is denoted $\mathbf{V}_c^b = (\omega_c^b, \mathbf{v}_c^b)^T$ with $\tilde{\omega}_c^b = \mathbf{R}_c^T \dot{\mathbf{R}}_c$, $\mathbf{v}_c^b = \mathbf{R}_c^T \dot{\mathbf{r}}_c$. The inertia matrix with respect to this COM frame is denoted

$$\mathbf{M}_c^b = \begin{pmatrix} \Theta_c & \mathbf{0} \\ \mathbf{0} & m\mathbf{I} \end{pmatrix} \tag{40}$$

with the body mass m and the inertia tensor Θ_c expressed in the body-fixed COM frame \mathcal{F}_c . Let $S_{bc} = (\mathbf{R}_{bc}, {}^b\mathbf{d}_{bc}) \in SE(3)$ be the transformation from the COM frame \mathcal{F}_c to the BFR \mathcal{F}_b . Here ${}^b\mathbf{d}_{bc}$ is the position vector from the BFR to the COM resolved in the BFR. Then the configuration of \mathcal{F}_c is given in terms of that of the BFR by $\mathbf{C}_c = \mathbf{C} S_{bc}$. The inertia matrix

with respect to the general BFR \mathcal{F}_b is

$$\begin{aligned} \mathbf{M}^b &= \mathbf{Ad}_{S_{bc}}^{-T} \mathbf{M}_c^b \mathbf{Ad}_{S_{bc}}^{-1} \\ &= \begin{pmatrix} \boldsymbol{\Theta}_{b_{\tilde{d}_{bc}}} & m^b \tilde{\mathbf{d}}_{bc} \\ -m^b \tilde{\mathbf{d}}_{bc} & m \mathbf{I} \end{pmatrix} \end{aligned} \tag{41}$$

with $\boldsymbol{\Theta}_b = \mathbf{R}_{bc} \boldsymbol{\Theta}_c \mathbf{R}_{bc}^T - m \tilde{\mathbf{d}}_{bc}^2$ (which is the parallel axes theorem).

The momentum coscrew represented in the body-fixed RFR \mathcal{F}_b is $\boldsymbol{\Pi}^b = \mathbf{M}^b \mathbf{V}^b$. The frame transformation of (39) to the BFR \mathcal{F}_b yields the body-fixed momentum balance represented in \mathcal{F}_b in the concise form:

$$\begin{aligned} \mathbf{W}^b &= \dot{\boldsymbol{\Pi}}^b - \mathbf{ad}_{\mathbf{V}^b}^T \boldsymbol{\Pi}^b \\ &= \mathbf{M}^b \dot{\mathbf{V}}^b - \mathbf{ad}_{\mathbf{V}^b}^T \mathbf{M}^b \mathbf{V}^b \end{aligned} \tag{42}$$

with the applied wrench $\mathbf{W}^b = (\mathbf{t}^b, \mathbf{f}^b)^T$ in body-fixed representation. Equations (42) are formally identical to the spatial equations (39). Written separately, this yields the NE equations expressed in an arbitrary body-fixed BFR

$$\boldsymbol{\Theta}_b \dot{\boldsymbol{\omega}}^b + \tilde{\boldsymbol{\omega}}^b \boldsymbol{\Theta}_b \boldsymbol{\omega}^b + m^b \tilde{\mathbf{d}}_{bc} (\dot{\mathbf{v}}^b + \tilde{\boldsymbol{\omega}}^b \mathbf{v}^b) = \mathbf{t}^b, \tag{43}$$

$$m (\dot{\mathbf{v}}^b + \tilde{\boldsymbol{\omega}}^b \mathbf{v}^b + (\tilde{\boldsymbol{\omega}}^b + \tilde{\boldsymbol{\omega}}^b \tilde{\boldsymbol{\omega}}^b)^b \mathbf{d}_{bc}) = \mathbf{f}^b. \tag{44}$$

When using the COM frame as special case, the momentum represented in the body-fixed COM frame is $\boldsymbol{\Pi}_c^b = \mathbf{M}_c^b \mathbf{V}_c^b$, and the momentum balance yields

$$\mathbf{W}_c^b = \mathbf{M}_c^b \dot{\mathbf{V}}_c^b - \mathbf{ad}_{\mathbf{V}_c^b}^T \mathbf{M}_c^b \mathbf{V}_c^b. \tag{45}$$

Written in components, this yields the NE equations represented in the COM frame:

$$\boldsymbol{\Theta}_c \dot{\boldsymbol{\omega}}_c^b + \tilde{\boldsymbol{\omega}}_c^b \boldsymbol{\Theta}_c \boldsymbol{\omega}_c^b = \mathbf{t}_c^b, \tag{46}$$

$$m (\dot{\mathbf{v}}_c^b + \tilde{\boldsymbol{\omega}}_c^b \mathbf{v}_c^b) = \mathbf{f}_c^b. \tag{47}$$

Noticeably, the angular and translational momentum equations are coupled even though the COM is used as reference. This is due to using body-fixed twists.

3.3 Hybrid form

The hybrid twist $\mathbf{V}_c^h = (\boldsymbol{\omega}^s, \dot{\mathbf{r}}_c)^T$ of the COM frame is related to the body-fixed twist by $\mathbf{Ad}_{\mathbf{R}_c}^{-1} \mathbf{V}_c^b$ (see (99)), where \mathbf{R}_c is the absolute rotation matrix of \mathcal{F}_c in C_c . The hybrid momentum screw is thus $\boldsymbol{\Pi}_c^h = \mathbf{M}_c^h \mathbf{V}_c^h$, where the hybrid representation of the inertia matrix is

$$\mathbf{M}_c^h = \mathbf{Ad}_{\mathbf{R}_c}^{-T} \mathbf{M}_c^b \mathbf{Ad}_{\mathbf{R}_c}^{-1} = \begin{pmatrix} \boldsymbol{\Theta}_c^h & \mathbf{0} \\ \mathbf{0} & m \mathbf{I} \end{pmatrix}, \quad \boldsymbol{\Theta}_c^h = \mathbf{R}_c \boldsymbol{\Theta}_c \mathbf{R}_c^T. \tag{48}$$

The hybrid momentum balance with respect to the COM follows from $\dot{\boldsymbol{\Pi}}_c^h = \mathbf{W}_c^h$. Using $\dot{\mathbf{M}}_c^h = -\mathbf{ad}_{\boldsymbol{\omega}^s}^T \mathbf{M}_c^h - \mathbf{M}_c^h \mathbf{ad}_{\boldsymbol{\omega}^s}$ yields

$$\mathbf{W}_c^h = \mathbf{M}_c^h \dot{\mathbf{V}}_c^h + \mathbf{ad}_{\boldsymbol{\omega}^s} \mathbf{M}_c^h \mathbf{V}_c^h \tag{49}$$

with $\mathbf{V}_c^h = (\boldsymbol{\omega}^s, \mathbf{0})^T$ (notice that $-\mathbf{ad}_{\boldsymbol{\omega}^s}^T = \mathbf{ad}_{\boldsymbol{\omega}^s}$). Writing (49) separately for the angular and linear momentum balance,

$$\boldsymbol{\Theta}_c^h \dot{\boldsymbol{\omega}}^s + \tilde{\boldsymbol{\omega}}^s \boldsymbol{\Theta}_c^h \boldsymbol{\omega}^s = \mathbf{t}_c^h, \tag{50}$$

$$m \dot{\mathbf{r}}_c = \mathbf{f}_c^h, \tag{51}$$

shows that the hybrid NE equations with respect to the COM are indeed decoupled. Here $\mathbf{W}_c^h = (\mathbf{t}_c^h, \mathbf{f}_c^h)^T$ denotes the hybrid wrench measured in the COM frame and resolved in the IFR.

Now consider the arbitrary body-fixed BFR \mathcal{F}_b with configuration $C = (\mathbf{R}, \mathbf{r})$. The hybrid twist $\mathbf{V}^h = (\boldsymbol{\omega}^s, \dot{\mathbf{r}})^T$ measured at this RFR is $\mathbf{V}^h = \mathbf{Ad}_{\mathbf{d}_{bc}} \mathbf{V}_c^h$, with the displacement vector \mathbf{d}_{bc} from BFR to COM resolved in the IFR. The hybrid mass matrix with respect to the BFR \mathcal{F}_b is found as

$$\mathbf{M}^h = \mathbf{Ad}_{\mathbf{d}_{bc}}^{-T} \mathbf{M}_c^h \mathbf{Ad}_{\mathbf{d}_{bc}}^{-1} = \begin{pmatrix} \boldsymbol{\Theta}^h & m \tilde{\mathbf{d}}_{bc} \\ -m \tilde{\mathbf{d}}_{bc} & m \mathbf{I} \end{pmatrix}, \quad \boldsymbol{\Theta}^h = \boldsymbol{\Theta}_c^h - m \tilde{\mathbf{d}}_{bc}^2. \tag{52}$$

The momentum balance in hybrid representation with respect to an arbitrary BFR

$$\dot{\mathbf{P}}^h = \mathbf{W}^h \tag{53}$$

is found, using $\dot{\mathbf{A}}_{\mathbf{d}_{bc}}^{-1} = -\mathbf{ad}_{\dot{\mathbf{d}}_{bc}} = \mathbf{Ad}_{\mathbf{d}_{bc}}^{-1} \mathbf{ad}_{\boldsymbol{\omega}^s} - \mathbf{ad}_{\boldsymbol{\omega}^s} \mathbf{Ad}_{\mathbf{d}_{bc}}^{-1}$ to evaluate (53), as

$$\boxed{\mathbf{W}^h = \mathbf{M}^h \dot{\mathbf{V}}^h + \mathbf{ad}_{\boldsymbol{\omega}^s} \mathbf{M}^h \mathbf{V}^h.} \tag{54}$$

Separating the angular and translational part results in

$$\boldsymbol{\Theta}^h \dot{\boldsymbol{\omega}}^s + \tilde{\boldsymbol{\omega}}^s \boldsymbol{\Theta}^h \boldsymbol{\omega}^s + m \tilde{\mathbf{d}}_{bc} \dot{\mathbf{r}} = \mathbf{t}^h, \tag{55}$$

$$m (\dot{\mathbf{r}} + (\tilde{\boldsymbol{\omega}}^s + \tilde{\boldsymbol{\omega}}^s \boldsymbol{\omega}^s) \mathbf{d}_{bc}) = \mathbf{f}^h. \tag{56}$$

These are simpler than the body-fixed equations (43) and (44). Finally, notice that $\mathbf{f}^h = \mathbf{f}^s$.

3.4 Mixed form

The mixed twist $\mathbf{V}^m = (\boldsymbol{\omega}^b, \dot{\mathbf{r}})^T$ consist of the body-fixed angular velocity $\boldsymbol{\omega}^b$, that is, measured and resolved in the BFR \mathcal{F}_b , and the translational velocity $\dot{\mathbf{r}}$ measured at the BFR \mathcal{F}_b and resolved in the IFR. The NE equations for the mixed representation with respect to a general BFR are directly found by combining (43) and (56) with $\dot{\mathbf{r}} = \dot{\mathbf{v}}^b + \tilde{\boldsymbol{\omega}}^b \mathbf{v}^b$:

$$\boldsymbol{\Theta}^b \dot{\boldsymbol{\omega}}^b + \tilde{\boldsymbol{\omega}}^b \boldsymbol{\Theta}^b \boldsymbol{\omega}^b + m^b \tilde{\mathbf{d}}_{bc} \mathbf{R}^T \dot{\mathbf{r}} = \mathbf{t}^b, \tag{57}$$

$$m (\dot{\mathbf{r}} + \mathbf{R} (\tilde{\boldsymbol{\omega}}^b + \tilde{\boldsymbol{\omega}}^b \boldsymbol{\omega}^b) \mathbf{d}_{bc}) = \mathbf{f}^h. \tag{58}$$

If a COM frame is used, then combining (46) and (51) yields

$$\begin{aligned} \boldsymbol{\Theta}_c \dot{\boldsymbol{\omega}}_c^b + \tilde{\boldsymbol{\omega}}_c^b \boldsymbol{\Theta}_c \boldsymbol{\omega}_c^b &= \mathbf{t}_c^b, \\ m \ddot{\mathbf{r}}_c &= \mathbf{f}_c^h. \end{aligned} \tag{59}$$

3.5 Arbitrary representation

The NE equations of body i represented in an arbitrary frame \mathcal{F}_j are obtained by a frame transformation of the spatial momentum balance (37) as

$$\mathbf{Ad}_{C_j}^T \dot{\mathbf{\Pi}}_i^s = \mathbf{Ad}_{C_j}^T \mathbf{W}_i^s. \tag{60}$$

The spatial twist in terms of the twist of body i represented in \mathcal{F}_j is $\mathbf{V}_i^s = \mathbf{Ad}_{C_j}^j \mathbf{V}_i$. Using $\dot{\mathbf{V}}_i^s = \mathbf{Ad}_{C_j}^j \dot{\mathbf{V}}_i + \mathbf{ad}_{\mathbf{V}_j^s} \mathbf{V}_i^s$, (39) yields

$${}^j \mathbf{M}_i ({}^j \dot{\mathbf{V}}_i + \mathbf{ad}_{\mathbf{V}_j^s} {}^j \mathbf{V}_i) - \mathbf{ad}_{\mathbf{V}_j^s}^T {}^j \mathbf{M}_i {}^j \mathbf{V}_i = {}^j \mathbf{W}_i \tag{61}$$

with the inertia matrix of body i represented in frame j

$${}^j \mathbf{M}_i := \mathbf{Ad}_{C_j}^T \mathbf{M}_i^s \mathbf{Ad}_{C_j}. \tag{62}$$

The spatial and body-fixed representations are particular cases with $i = j$.

Even more generally, the NE equations can be resolved in yet another frame \mathcal{F}_k . This is achieved by transforming the momentum balance (37) as

$$\mathbf{Ad}_{R_{j,k}}^T \mathbf{Ad}_{C_j}^T \dot{\mathbf{\Pi}}_i^s = \mathbf{Ad}_{R_{j,k}}^T \mathbf{Ad}_{C_j}^T \mathbf{W}_i^s \tag{63}$$

where $\mathbf{R}_{k,j}$ is the rotation matrix from \mathcal{F}_i to \mathcal{F}_k . The final equations follow from (61) and the relation ${}^j \dot{\mathbf{V}}_i^j = \mathbf{Ad}_{R_{j,k}} {}^k \dot{\mathbf{V}}_i^j + \mathbf{ad}_{\mathbf{V}_k^j} \mathbf{Ad}_{R_{j,k}} {}^k \mathbf{V}_i^j$ as

$$\boxed{{}^k \mathbf{M}_i^j ({}^k \dot{\mathbf{V}}_i^j + (\mathbf{ad}_{\mathbf{V}_k^j} + \mathbf{ad}_{\mathbf{V}_k^j}) {}^k \mathbf{V}_i^j) - \mathbf{ad}_{\mathbf{V}_k^j}^T {}^k \mathbf{M}_i^j {}^k \mathbf{V}_i^j = {}^k \mathbf{W}_i^j} \tag{64}$$

with the mass matrix of body i measured at frame \mathcal{F}_j and resolved in frame \mathcal{F}_k

$${}^k \mathbf{M}_i^j := \mathbf{Ad}_{R_{k,j}}^T {}^j \mathbf{M}_i \mathbf{Ad}_{R_{k,j}} = \mathbf{Ad}_{R_{k,j}}^T \mathbf{Ad}_{C_j}^T \mathbf{M}_i^s \mathbf{Ad}_{C_j} \mathbf{Ad}_{R_{k,j}}. \tag{65}$$

The spatial and body-fixed representations are particular cases with $i = j = k$, and the hybrid representation with $i = j$ and $k = 0$. An alternative form of the NE equations in arbitrary reference frames was presented in [9].

4 Recursive evaluation of the motion equations for a kinematic chain

The model-based control of complex MBS and the computational MBS dynamics rely on efficient recursive inverse and forward dynamics algorithms. The recursive Newton–Euler method for tree-topology MBS was presented in an abstract, that is, coordinate-free, approach in [46]. However, the various recursive methods using different representations give rise to algorithmically equivalent methods but with different computational costs. In the following, the various inverse dynamics algorithms are presented, and their computational effort is estimated. A detailed analysis and forward dynamics algorithms are beyond the scope of this paper. The presented discussion is nevertheless indicative also for the corresponding forward dynamics algorithms. Some results on the forward kinematics complexity can be

found in [65, 79, 85]. This, however, depends on the actual implementation. A comparative study is still due and will be part of further research.

The inverse dynamics consists in evaluating the motion equations for given joint coordinates \mathbf{q} , joint rates $\dot{\mathbf{q}}$, accelerations $\ddot{\mathbf{q}}$, and applied wrenches $\mathbf{W}_i^{\text{app}}$ and hence in determining the joint forces $\mathbf{Q} = (Q_1, \dots, Q_n)$. The starting point of recursive algorithms for rigid body MBS is the NE equations of the individual bodies. The MBS dynamics is indeed governed by the Lagrange equations. Consequently, summarizing the recursive steps yields the Lagrangian motion equations in closed form. This will be shown in the following.

It is assumed for simplicity that the inertia properties, that is, the mass matrices \mathbf{M}_i^b , are expressed in the body-fixed BFR of body i determining its configuration, rather than introducing a second frame.

4.1 Body-fixed representation

Forward kinematics recursion Given the joint variables \mathbf{q} , the configurations of the n bodies are determined recursively by (93) or (94), and the twists by (109). Then also the accelerations are found recursively. The expression $\mathbf{C}_{i-1,i}(q_i) = \mathbf{B}_i \exp({}^i\mathbf{X}_i q_i)$ for the relative configuration yields $\text{Ad}_{\mathbf{C}_{i,i-1}} \mathbf{V}_{i-1}^b = [\text{Ad}_{\mathbf{C}_{i,i-1}} \mathbf{V}_{i-1}^b, {}^i\mathbf{X}_i \dot{q}_i]$, and hence

$$\dot{\mathbf{V}}_i^b = \text{Ad}_{\mathbf{C}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^b + [\text{Ad}_{\mathbf{C}_{i,i-1}} \mathbf{V}_{i-1}^b, {}^i\mathbf{X}_i \dot{q}_i] + {}^i\mathbf{X}_i \ddot{q}_i \tag{66a}$$

$$= \text{Ad}_{\mathbf{C}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^b + [\text{Ad}_{\mathbf{C}_{i,i-1}} \mathbf{V}_{i-1}^b, \mathbf{V}_i^b] + {}^i\mathbf{X}_i \ddot{q}_i \tag{66b}$$

$$= \text{Ad}_{\mathbf{C}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^b + [\mathbf{V}_i^b, {}^i\mathbf{X}_i \dot{q}_i] + {}^i\mathbf{X}_i \ddot{q}_i, \tag{66c}$$

where (66b) and (66c) follow by replacing either argument in the Lie bracket using (109).

Remark 3 Notice that solving (109) for \dot{q}_i leads to the result in Remark 9 of [60]. Solving (66c) for \ddot{q}_i yields (12). Using (66b), the latter can be expressed as $\ddot{q}_i = {}^i\mathbf{X}_i^T (\mathbf{V}_i^b - \text{Ad}_{\mathbf{C}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^b + [\mathbf{V}_i^b, \text{Ad}_{\mathbf{C}_{i,i-1}} \mathbf{V}_{i-1}^b]) / \|{}^i\mathbf{X}_i\|^2$.

Recursive Newton–Euler algorithm Once the configurations, twists, and accelerations of the bodies are computed with the forward kinematics recursion, the Newton–Euler equations (42) for each individual body can be evaluated by an inverse dynamics backward recursion. The momentum balance of body i then yields the resulting body-fixed wrench \mathbf{W}_i^b acting on the body due to generalized joint forces and constraint reactions forces. Projecting the resultant wrench onto the screw axis ${}^i\mathbf{X}_i$ of joint i yields the generalized force Q_i . Summarizing the forward and backward recursions yields the following recursive algorithm:

Forward Kinematics

- Input: $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$
- For $i = 1, \dots, n$

$$\mathbf{C}_i = \mathbf{C}_{i-1} \mathbf{B}_i \exp({}^i\mathbf{X}_i q_i) = \exp(\mathbf{Y}_1 q_1) \cdot \dots \cdot \exp(\mathbf{Y}_i q_i) \mathbf{A}_i \tag{67a}$$

$$\mathbf{V}_i^b = \text{Ad}_{\mathbf{C}_{i,i-1}} \mathbf{V}_{i-1}^b + {}^i\mathbf{X}_i \dot{q}_i \tag{67b}$$

$$\dot{\mathbf{V}}_i^b = \text{Ad}_{\mathbf{C}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^b - \dot{q}_i \text{ad}_{{}^i\mathbf{X}_i} \mathbf{V}_i^b + {}^i\mathbf{X}_i \ddot{q}_i \tag{67c}$$

- Output: $\mathbf{C}_i, \mathbf{V}_i^b, \dot{\mathbf{V}}_i^b$

Inverse Dynamics

- Input: $\mathbf{C}_i, \mathbf{V}_i^b, \dot{\mathbf{V}}_i^b, \mathbf{W}_i^{b,app}$
- For $i = n - 1, \dots, 1$

$$\mathbf{W}_i^b = \mathbf{Ad}_{\mathbf{C}_{i+1,i}}^T \mathbf{W}_{i+1}^b + \mathbf{M}_i^b \dot{\mathbf{V}}_i^b - \mathbf{ad}_{\mathbf{V}_i^b}^T \mathbf{M}_i^b \mathbf{V}_i^b + \mathbf{W}_i^{b,app} \tag{68a}$$

$$\mathbf{Q}_i = {}^i \mathbf{X}_i^T \mathbf{W}_i^b \tag{68b}$$

- Output: \mathbf{Q}

The joint reaction wrench is omitted in (68a) since this is reciprocal to the joint screw and does not contribute to (68b). Notice that, with (95), the body-fixed ${}^i \mathbf{X}_i$ and the spatial representation \mathbf{Y}_i of joint screw coordinates can be used. This form of the recursive body-fixed NE equations, using Lie group notation, has been reported in several publications [50, 67, 70, 71].

Computational effort For the kinematic chain comprising n bodies connected by $n - 1$ DOF joints, in total, the twist recursion (67b) and acceleration recursion (67c) each requires $n - 1$ frame transformations. The acceleration recursion (67c) further requires $n - 1$ Lie brackets. The second argument of the Lie bracket can be reused from (67b). Hence the twist and acceleration recursion need $2(n - 1)$ frame transformations and $n - 1$ Lie brackets. The backward recursion (68a) needs $n - 1$ frame transformations and n Lie brackets. In total, the NE algorithm needs $3(n - 1)$ frame transformations and $2n - 1$ Lie brackets. The evaluation of the Lie bracket in (67c) can be simplified using (66b) since the screw vector ${}^i \mathbf{X}_i$ expressed in RFR is sparse and often only contains one non-zero entry.

Remark on forward dynamics Using the body-fixed representation, a recursive forward dynamics algorithm, making explicit use of Lie group concepts, was presented in [67, 68, 70, 71, 76]. The kinematic forward recursion, together with the factorization in Sect. 3.1.3 of [60], was used to derive $O(n)$ forward dynamics algorithms in [30, 44], where the Lie group concept is regarded as spatial operator algebra. Other $O(n)$ forward dynamics algorithms were presented in [2, 7, 34, 35]. The inverse dynamics formulation was also presented in [29, 45] in the context of screw theory.

4.2 Spatial representation

Forward kinematics recursion Expressing the spatial twist in terms of the spatial Jacobian, the expressions (105) lead immediately to

$$\mathbf{V}_i^s = \mathbf{V}_{i-1}^s + \mathbf{J}_i^s \dot{q}_i. \tag{69}$$

The recursive determination of spatial accelerations thus only requires the time derivative (17) of the spatial Jacobian, so that

$$\begin{aligned} \dot{\mathbf{V}}_i^s &= \dot{\mathbf{V}}_{i-1}^s + \mathbf{ad}_{\mathbf{V}_{i-1}^s} \mathbf{J}_i^s \dot{q}_i + \mathbf{J}_i^s \ddot{q}_i \\ &= \dot{\mathbf{V}}_{i-1}^s + \mathbf{ad}_{\mathbf{V}_{i-1}^s} \mathbf{V}_i^s + \mathbf{J}_i^s \ddot{q}_i. \end{aligned} \tag{70}$$

The second form in (70) follows by inserting (69). This is a generalization of Euler’s theorem, for the derivative of vectors resolved in moving frames, to screw coordinate vectors. Therefore the \mathbf{ad} operator is occasionally called the “spatial cross product”.

Recursive Newton–Euler algorithm The momentum balance expressed with the spatial NE equations (39), together with (69), leads to the following algorithm:

Forward Kinematics

- Input: $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$
- For $i = 1, \dots, n$

$$\mathbf{C}_i = \mathbf{C}_{i-1} \mathbf{B}_i \exp({}^i \mathbf{X}_i q_i) = \exp(\mathbf{Y}_1 q_1) \cdot \dots \cdot \exp(\mathbf{Y}_i q_i) \mathbf{A}_i \tag{71a}$$

$$\mathbf{J}_i^s = \mathbf{Ad}_{\mathbf{C}_i} {}^i \mathbf{X}_i = \mathbf{Ad}_{\mathbf{C}_i \mathbf{A}_i^{-1}} \mathbf{Y}_i = \mathbf{Ad}_{\mathbf{C}_j \mathbf{S}_{j,j}^{j-1}} \mathbf{Z}_j \tag{71b}$$

$$\mathbf{V}_i^s = \mathbf{V}_{i-1}^s + \mathbf{J}_i^s \dot{q}_i \tag{71c}$$

$$\dot{\mathbf{V}}_i^s = \dot{\mathbf{V}}_{i-1}^s + \mathbf{J}_i^s \ddot{q}_i + \mathbf{ad}_{\mathbf{V}_{i-1}^s} \mathbf{V}_i^s \tag{71d}$$

- Output: $\mathbf{C}_i, \mathbf{V}_i^s, \dot{\mathbf{V}}_i^s, \mathbf{J}_i^s$

Inverse Dynamics

- Input: $\mathbf{C}_i, \mathbf{V}_i^s, \dot{\mathbf{V}}_i^s, \mathbf{J}_i^s, \mathbf{W}_i^{s,app}$
- For $i = n - 1, \dots, 1$

$$\mathbf{M}_i^s = \mathbf{Ad}_{\mathbf{C}_i}^{-T} \mathbf{M}_i^b \mathbf{Ad}_{\mathbf{C}_i}^{-1} \tag{72a}$$

$$\mathbf{W}_i^s = \mathbf{W}_{i+1}^s + \mathbf{M}_i^s \dot{\mathbf{V}}_i^s - \mathbf{ad}_{\dot{\mathbf{V}}_i^s}^T \mathbf{M}_i^s \mathbf{V}_i^s + \mathbf{W}_i^{s,app} \tag{72b}$$

$$\mathbf{Q}_i = (\mathbf{J}_i^s)^T \mathbf{W}_i^s \tag{72c}$$

- Output: \mathbf{Q}

Computational effort In contrast to (67b), once the instantaneous screws (71b) and the spatial mass matrix (35) are computed, recursions (71c), (71d), and (72b) do not require frame transformations of twists. Instead, the spatial mass matrix is transformed according to (72a), which is the frame transformation of a second-order tensor. Overall, the spatial algorithm needs n frame transformations of screw coordinates, n frame transformation of a second-order tensor, and $2n - 1$ Lie brackets. Comparing body-fixed and spatial formulation, it must be noticed that the frame transformation of the second-order inertia tensor has the same complexity as two screw coordinate transformations (if just implemented in the form (35)), and hence the computational complexity of both would be equivalent. This fact is to be expected since body-fixed and spatial representations are related by frame transformations. Nevertheless, the spatial version has some interesting features that should be emphasized:

1. The NE equations (39) form a nonlinear first-order ODE system on $SE(3) \times se(3)$. Since a spatial reference is used, the momentum conservation of a rigid body can simply be written as $\dot{\mathbf{\Pi}}_i^s = \mathbf{0}$, where $\mathbf{\Pi}_i^s \in se^*(3)$ is the momentum coscrew. Using the spatial momentum balance (37) has potentially two advantages. Firstly, (37) is a linear ODE in $\mathbf{\Pi}$ on the phase space $SE(3) \times se^*(3)$. This implies that a numerical integration scheme can easily preserve the momentum, as pointed out in [12]. Secondly, $O(n)$ formulations using canonical momenta have been shown to be computationally advantageous. An $O(n)$ forward dynamics algorithm based on the canonical Hamilton equations was presented in [64] that uses a hybrid form. It was shown to require less numerical operations than $O(n)$ algorithms based on the NE equations. It is also known that $O(n)$ algorithms based on the spatial representation can be computationally more efficient than those based on

- body-fixed or hybrid representations [29]. A further reduction of computational costs can be expected from an algorithm using spatial momenta.
2. It is interesting to notice that the hybrid as well as the spatial twists appear in the recursive $O(n)$ forward dynamics formulation in [6], where the first is called “Cartesian velocity”, and the latter is called “velocity state”. In this formulation the spatial twist plays a central role, and it was already remarked that the recursive relation of spatial twists (see (71c)) is simpler than that for hybrid twists (76c).
 3. If a *purely kinematic analysis* is envisaged the forward recursion (71b)–(71d) is more efficient than the body-fixed and the hybrid version (see the next section) [65] (disregarding possibly necessary transformations of the results to local reference frames). As pointed out in Sect. 2.2, this advantage is retained for the higher-order kinematics (jerk, jounce, etc.) [54].

Remark on forward dynamics The spatial formulation is rarely used for dynamics. Featherstone [26, 29] derived a forward dynamics $O(n)$ algorithm. He concluded that this requires the lowest computational effort compared to other methods. But this does not take into account the necessary transformations of twists and wrenches to local reference frames. Moreover, it was shown in [79] that the $O(n)$ forward dynamics algorithm in body-fixed representation, using the body-fixed joint screw coordinates ${}^i\mathbf{X}_i$ and RFR at the joint axis, can be implemented in such a way that it requires less computational effort than the spatial version. The key is that when the BFR of \mathcal{F}_i is located at and aligned with the axis of joint i , then ${}^i\mathbf{X}_i$ becomes sparse. From a user perspective this, however, is a restraining presumption.

4.3 Hybrid form

Forward kinematics recursion The hybrid twists are determined recursively by (111) with ${}^0\mathbf{X}_i^j = \mathbf{Ad}_{\mathbf{R}_i} {}^i\mathbf{X}_i$. For the acceleration recursion, note that $\mathbf{Ad}_{\mathbf{r}_{i,i-1}} = \mathbf{ad}_{\dot{\mathbf{r}}_{i,i-1}} = \mathbf{ad}_{\dot{\mathbf{r}}_{i-1}} - \mathbf{ad}_{\dot{\mathbf{r}}_i}$ since $\dot{\mathbf{r}}_{i,i-1} = \dot{\mathbf{r}}_{i-1} - \dot{\mathbf{r}}_i$. This yields

$$\dot{\mathbf{V}}_i^h = \mathbf{Ad}_{\mathbf{r}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^h + {}^0\mathbf{X}_i^i \ddot{q}_i + \mathbf{ad}_{\dot{\mathbf{r}}_{i,i-1}} \mathbf{V}_{i-1}^h + \mathbf{ad}_{\omega_i} {}^0\mathbf{X}_i^i \dot{q}_i. \tag{73}$$

Taking into account that $\mathbf{ad}_{\dot{\mathbf{r}}_i} (\mathbf{V}_i^h - \mathbf{X}_i^h \dot{q}_i) = \mathbf{ad}_{\dot{\mathbf{r}}_i} \mathbf{V}_{i-1}^h$ (because there is no angular part in $\mathbf{ad}_{\dot{\mathbf{r}}_i}$) and $\mathbf{ad}_{\dot{\mathbf{r}}_i} + \mathbf{ad}_{\omega_i^s} = \mathbf{ad}_{\mathbf{v}_i^h}$, this can be transformed to

$$\dot{\mathbf{V}}_i^h = \mathbf{Ad}_{\mathbf{r}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^h + {}^0\mathbf{X}_i^i \ddot{q}_i + \mathbf{ad}_{\dot{\mathbf{r}}_{i-1}} \mathbf{V}_{i-1}^h - \mathbf{ad}_{\dot{\mathbf{r}}_i} \mathbf{V}_i^h + \mathbf{ad}_{\mathbf{v}_i^h} {}^0\mathbf{X}_i^i \dot{q}_i. \tag{74}$$

Another form follows by solving (111) for \mathbf{V}_{i-1}^h and inserting this into (73), noting that $\mathbf{ad}_{\dot{\mathbf{r}}_{i,i-1}} \mathbf{Ad}_{\mathbf{r}_{i,i-1}}^{-1} = \mathbf{ad}_{\dot{\mathbf{r}}_{i-1}}$, as

$$\dot{\mathbf{V}}_i^h = \mathbf{Ad}_{\mathbf{r}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^h + {}^0\mathbf{X}_i^i \ddot{q}_i + \mathbf{ad}_{\dot{\mathbf{r}}_{i-1}} \mathbf{V}_i^h + (\mathbf{ad}_{\mathbf{v}_i^h} - \mathbf{ad}_{\dot{\mathbf{r}}_{i-1}}) {}^0\mathbf{X}_i^i \dot{q}_i. \tag{75}$$

Comparing these three different recursive relations (73), (74), and (75) for the hybrid acceleration from a computational perspective (73) is the most efficient.

Recursive Newton–Euler algorithm With the hybrid Newton–Euler equations (54), the recursive NE algorithm is as follows:

Forward Kinematics

- Input: $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$
- For $i = 1, \dots, n$

$$\mathbf{C}_i = \mathbf{C}_{i-1}(\mathbf{q})\mathbf{B}_i \exp({}^i\mathbf{X}_i q_i) = \exp(\mathbf{Y}_1 q_1) \cdot \dots \cdot \exp(\mathbf{Y}_i q_i)\mathbf{A}_i \tag{76a}$$

$${}^0\mathbf{X}_i^i = \mathbf{Ad}_{\mathbf{R}_i} {}^i\mathbf{X}_i = \mathbf{Ad}_{-\mathbf{r}_i} \mathbf{Y}_i = \mathbf{Ad}_{\mathbf{R}_i} \mathbf{Ad}_{\mathbf{S}_{i,i}} {}^{i-1}\mathbf{Z}_i \tag{76b}$$

$$\mathbf{V}_i^h = \mathbf{Ad}_{\mathbf{r}_{i,i-1}} \mathbf{V}_{i-1}^h + {}^0\mathbf{X}_i^i \dot{q}_i \tag{76c}$$

$$\dot{\mathbf{V}}_i^h = \mathbf{Ad}_{\mathbf{r}_{i,i-1}} \dot{\mathbf{V}}_{i-1}^h + \mathbf{ad}_{\mathbf{r}_{i,i-1}} \mathbf{V}_{i-1}^h + \mathbf{ad}_{\omega_i^s} {}^0\mathbf{X}_i^i \dot{q}_i + {}^0\mathbf{X}_i^i \ddot{q}_i \tag{76d}$$

- Output: $\mathbf{C}_i, \mathbf{V}_i^h, \dot{\mathbf{V}}_i^h, {}^0\mathbf{X}_i^i$

Inverse Dynamics

- Input: $\mathbf{C}_i, \mathbf{V}_i^h, \dot{\mathbf{V}}_i^h, {}^0\mathbf{X}_i^i, \mathbf{W}_i^{\text{h,app}}$
- For $i = n - 1, \dots, 1$

$$\mathbf{M}_i^h = \mathbf{Ad}_{\mathbf{R}_i}^{-T} \mathbf{M}_i^b \mathbf{Ad}_{\mathbf{R}_i} \tag{77a}$$

$$\mathbf{W}_i^h = \mathbf{Ad}_{\mathbf{d}_{i+1,i}}^T \mathbf{W}_{i+1}^h + \mathbf{M}_i^h \dot{\mathbf{V}}_i^h + \mathbf{ad}_{\omega_i^h} \mathbf{M}_i^h \dot{\mathbf{V}}_i^h + \mathbf{W}_i^{\text{h,app}} \tag{77b}$$

$$\mathbf{Q}_i = ({}^0\mathbf{X}_i^i)^T \mathbf{W}_i^h \tag{77c}$$

- Output: \mathbf{Q}

Computational effort The hybrid representation is a compromise between using twists and wrenches measured in body-fixed frames (as for the body-fixed representation, where twists and wrenches are measured at the RFR origin) and those resolved in the IFR (as for the spatial representation, where twists and wrenches are measured at the IFR origin). It has therefore been used extensively for $O(n)$ inverse and forward dynamics algorithms. The essential difference between the forward recursion for kinematic evaluation in body-fixed and hybrid formulation is that the body-fixed recursion (67a)–(67c) requires frame transformations of screws involving rotations and translations, whereas the hybrid recursion (76a)–(76d) only requires the change of reference point using position vectors resolved in the IFR. The attitude transformation only appears in (76b) and in the computation of the hybrid inertia matrix (77a). In total, the forward kinematics needs n rotational transformations and $2n - 2$ translational transformations. Further, (76d) needs $n - 1$ cross products of the form $\mathbf{ad}_{\mathbf{r}_{i,i-1}} \mathbf{V}_{i-1}^h = (\mathbf{0}, (\mathbf{r}_{i-1} - \mathbf{r}_i) \times \omega_{i-1}^s)^T$ and n Lie brackets $\mathbf{ad}_{\omega_i^s} {}^0\mathbf{X}_i^i$. The inverse dynamics needs the n rotational transformations (77a) of the second-order inertia tensor, $n - 1$ translational transformations of wrenches, and n Lie brackets with ω_i^s in (77b). In total, the hybrid NE algorithm needs $3n - 3$ translational and n rotational transformations of screw coordinates, n rotational transformations of the inertia tensor, and $3n - 1$ Lie brackets. Although the number of operations is equivalent to the body-fixed version, the particular form of transformations is computationally very simple motivating its extensive use in $O(n)$ forward dynamics algorithms. Moreover, the hybrid NE equations are commonly expressed in a body-fixed BFR at the COM, so that the hybrid NE equations simplify to (49) and (50), (51), respectively.

Instead of transforming the joint screws ${}^i\mathbf{X}_i$ or \mathbf{Y}_i in the reference configuration, the instantaneous hybrid joint screws can be determined using the defining expression (36) in [60] with the current $\mathbf{b}_{j,i}$ and \mathbf{e}_j .

Remark on forward dynamics The above inverse dynamics formulation was presented in [37, 38, 73, 74] together with $O(n)$ forward dynamics algorithms. An $O(n)$ forward dynamics method was presented in [1, 6]. These algorithms are deemed efficient taking into account that the computation results do not have to be transformed to the body-fixed reference points of interest, as in case of the spatial version. An $O(n)$ forward dynamics algorithm was developed in [64] using canonical Hamilton equations in hybrid representation, that is, the momentum balance (53) in terms of the conjugate momenta Π_i^h , rather than the NE equations. It was concluded that its performance is comparable to that of Featherstone's [29] method in terms of spatial twists.

4.4 Choice of body-fixed reference frames

The Lie group formulation involves geometric and inertia properties that are readily available, for example, from CAD data.

In [60] and in the preceding sections, two approaches to the description of MBS geometry (with and without body-fixed joint frames) and three versions for representing velocities and accelerations (body-fixed, spatial, hybrid) were presented, each having its merits. The description of the geometry is independent from the representation of twists. For instance, the geometry could be described in terms of joint screws expressed in the IFR, whereas the kinematics and dynamics is modeled using body-fixed twists. This allows us to take advantage of the low-complexity hybrid or spatial recursive NE equations while still having the freedom to use or avoid body-fixed joint frames.

The standard approach to model an MBS is to introduce (1) an IFR, (2) body-fixed BFRs, and (3) body-fixed JFRs. The latter is avoided using spatial joint screws \mathbf{Y}_i , as already presented. It still remains to introduce body-fixed BFRs kinematically representing the bodies. However, even the *explicit* definition of RFRs can be avoided by properly placing them. Their location is usually dictated by the definition of the inertia tensors, and it is customary to relate the inertia data to the COM. If, instead, the body-fixed BFRs are assigned such that they coincide in the reference configuration ($\mathbf{q} = \mathbf{0}$) with the IFR, then no reference configurations of bodies need to be determined ($\mathbf{A}_i = \mathbf{I}$). This normally means that the RFR is outside the physical extension of a body, that is, the inertia properties of all bodies are determined in the assembly reference configuration with respect to the global IFR. In other words, they are deduced from the design drawing (corresponding to $\mathbf{q} = \mathbf{0}$) relative to a single construction frame. This can be exploited when using CAD systems. The required kinematic data then reduces to the direction and position vectors \mathbf{e}_i and \mathbf{y}_i in order to compute \mathbf{Y}_i in (95). As a side effect, $\mathbf{Y}_i = {}^i\mathbf{X}_i$. This is an important result and does apply to any of the discussed twist representations since the representation of twists has nothing to do with the geometry description. Moreover, then the POE (94) and the Jacobian (104), (105), and thus (107) simplify in terms of spatial screw coordinates. The only computational drawback is that the hybrid Newton and Euler equations are not decoupled since the spatial IFR, to which the inertia data is related, is unlikely to coincide with the COM of the bodies in the reference configuration. Details can be found in [53].

5 Motion equations in closed form

5.1 Euler–Jourdain equations

The body-fixed NE equations for the individual bodies within the MBS are

$$\mathbf{M}_i^b \dot{\mathbf{V}}_i^b - \mathbf{ad}_{\mathbf{V}_i^b}^T \mathbf{M}_i^b \mathbf{V}_i^b - \mathbf{W}_i^{\text{b,app}} - \mathbf{W}_i^{\text{b,c}} = \mathbf{0}, \quad (78)$$

where $\mathbf{W}_i^{b,c}$ is the constraint reaction wrench of joint i , and $\mathbf{W}_i^{b,app}$ represents the total wrench applied to body i including the applied wrench in joint i . Jourdain’s principle of virtual power, using the admissible variation $\delta \mathbf{V}^b = \mathbf{J}^b \delta \dot{\mathbf{q}}$ of the system twist and noting that $\delta \mathbf{V}^b$ are reciprocal to the constraint wrenches (see Sect. B.2), yields the system of n motion equations

$$(\mathbf{J}^b)^T \begin{pmatrix} \mathbf{M}_1^b \dot{\mathbf{V}}_1^b - \mathbf{ad}_{\mathbf{V}_1^b}^T \mathbf{M}_1^b \mathbf{V}_1^b - \mathbf{W}_1^{b,app} \\ \vdots \\ \mathbf{M}_n^b \dot{\mathbf{V}}_n^b - \mathbf{ad}_{\mathbf{V}_n^b}^T \mathbf{M}_n^b \mathbf{V}_n^b - \mathbf{W}_n^{b,app} \end{pmatrix} = \mathbf{0}. \tag{79}$$

This form allows for a concise and computationally efficient construction of the motion equations. The point of departure are the NE equations of the individual bodies. The body-fixed system Jacobian (113) is determined by the (constant) joint screw coordinates in \mathbb{X}^b and the screw transformations encoded in \mathbf{A}^b . The same applies to the other representations. The accelerations are determined by (7), respectively (11). Explicit evaluation of (79) leads to the recursive algorithm in Sect. 4.1. Inserting the twists and accelerations into (79) yields Eqs. (1) that determine the MBS dynamics on the tangent bundle $T\mathbb{V}^n$ with state vector $(\mathbf{q}, \dot{\mathbf{q}}) \in T\mathbb{V}^n$. Alternatively, combining (79) with (112) yields a system of $n + 6n$ ODEs in the state variables $(\mathbf{q}, \mathbf{V}^b) \in \mathbb{V}^n \times se(3)^n$ that govern the dynamics on the state space $\mathbb{V}^n \times se(3)^n$. The advantage of this formulation is that it is a first-order ODE system and that the system has block triangular structure. Yet another interesting formulation follows with the NE equations (37) in terms of the conjugate momenta in spatial representation

$$(\mathbf{J}^s)^T \begin{pmatrix} \dot{\mathbf{\Pi}}_1^s - \mathbf{w}_1^{s,app} \\ \vdots \\ \dot{\mathbf{\Pi}}_n^s - \mathbf{w}_n^{s,app} \end{pmatrix} = \mathbf{0}, \tag{80}$$

$$\mathbf{M}_i^s \mathbf{J}_i^s \dot{\mathbf{q}} = \mathbf{\Pi}_i^s, \quad i, \dots, n.$$

This is a system of $n + 6n$ first-order ODEs in the phase space $(\mathbf{q}, \mathbf{\Pi}^s) \in \mathbb{V}^n \times se^*(3)^n$. System (80) can be solved for the $\dot{\mathbf{\Pi}}_i^s$ and $\dot{\mathbf{q}}_i$ noting the block triangular structure of \mathbf{J}^s [60]. From a numerical point of view, the momentum formulation in phase space will allow for momentum-preserving integration schemes.

Various versions of (79) have been published. Using the hybrid representation of twists, basically the same equations were reported in [4]. There the system Jacobian is called the “natural orthogonal complement”, motivated by the fact that the columns of \mathbf{J}^b are orthogonal to the vectorial representations of constraint wrenches (although the former are screws, whereas the latter are coscrews). In classical vector notation, they were reported in [39, 48] and [14]. In [39] the equations (79) are called the Euler–Jourdain equations. In [48], emphasizing on the recursive evaluation of the body Jacobian, the instantaneous body-fixed joint screws \mathbf{J}_i^b are called “kinematic basic functions” since they are the intrinsic objects in MBS kinematics. In [14] Eqs. (79) are called “projection equations” since the NE equations of the individual bodies are restricted to the feasible motion (although \mathbf{J}^b is not a projector). Equations (79) in body-fixed representation are equivalent to Kane’s equations, where \mathbf{J}_i^b are called “partial velocities” [40]. The instantaneous joint screw coordinates, that is, the columns \mathbf{J}_i^b of the geometric Jacobian, were also called “kinematic influence coefficients” and their partial derivatives (6) the “second-order kinematic influence coefficients” [8, 82].

It should be finally remarked that, due to the block triangular form of \mathbf{J}^b , solving (79) and using the inversion of \mathbf{A}^b (see (25) in [60]) lead immediately to an $O(n)$ forward dynamics

algorithm. This is the common starting point for deriving forward dynamics algorithms that applies to any twist representation.

5.2 Lagrange equations

The MBS motion equations can be derived as the Lagrange equations in terms of generalized coordinates. For simplicity, potential forces are omitted, so that the Lagrangian is simply the kinetic energy. Then the equations attain the form

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\mathbf{q}}} \right)^T - \left(\frac{\partial T}{\partial \mathbf{q}} \right)^T = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\dot{\mathbf{q}}, \mathbf{q})\dot{\mathbf{q}} = \mathbf{Q}(\dot{\mathbf{q}}, \mathbf{q}, t) \tag{81}$$

with generalized mass matrix \mathbf{M} , where $\mathbf{C}(\dot{\mathbf{q}}, \mathbf{q})\dot{\mathbf{q}}$ represent the Coriolis and centrifugal forces. The vector \mathbf{Q} stands for all other generalized forces, including potential, dissipative, and applied forces. Using body-fixed twists, the kinetic energy of body i is $T_i = \frac{1}{2}(\mathbf{V}_i^b)^T \mathbf{M}_i^b \mathbf{V}_i^b$. The kinetic energy of the MBS is $T(\dot{\mathbf{q}}, \mathbf{q}) = \sum_i T_i = \frac{1}{2}(\mathbf{V}^b)^T \mathbf{M}^b \mathbf{V}^b = \frac{1}{2}\dot{\mathbf{q}}^T \mathbf{M}\dot{\mathbf{q}}$ with the generalized mass matrix

$$\boxed{\mathbf{M}(\mathbf{q}) = (\mathbf{J}^b)^T \mathbf{M}^b \mathbf{J}^b} \tag{82}$$

and $\mathbf{M}^b := \text{diag}(\mathbf{M}_1^b, \dots, \mathbf{M}_n^b)$. The conjugate momentum vector is thus $(\frac{\partial T}{\partial \dot{\mathbf{q}}})^T = (\mathbf{J}^b)^T \mathbf{M}^b \mathbf{V}^b$. Its time derivative with (11) is given as $\frac{d}{dt} (\frac{\partial T}{\partial \dot{\mathbf{q}}})^T = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} - (\mathbf{J}^b)^T ((\mathbf{M}^b \mathbf{A}^b \mathbf{a}^b)^T + \mathbf{M}^b \mathbf{A}^b \mathbf{a}^b) \mathbf{J}^b \dot{\mathbf{q}}$, and \mathbf{a}^b is defined in (10). From (8) it follows that $(\frac{\partial T}{\partial \mathbf{q}})^T = (\mathbf{M}^b \mathbf{A}^b \mathbf{b}^b \mathbf{X}^b)^T \mathbf{J}^b \dot{\mathbf{q}}$ with

$$\mathbf{b}^b(\mathbf{V}^b) := \text{diag}(\mathbf{ad}_{\mathbf{v}_1^b}, \dots, \mathbf{ad}_{\mathbf{v}_n^b}). \tag{83}$$

This admits identifying the generalized mass matrix (82) and the matrix

$$\begin{aligned} \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) &= -(\mathbf{J}^b)^T ((\mathbf{M}^b \mathbf{A}^b \mathbf{a}^b)^T + \mathbf{M}^b \mathbf{A}^b \mathbf{a}^b) \mathbf{J}^b - (\mathbf{M}^b \mathbf{A}^b \mathbf{b}^b \mathbf{X}^b)^T \mathbf{J}^b \\ &= -(\mathbf{a}^b \mathbf{J}^b + \mathbf{b}^b \mathbf{X}^b)^T (\mathbf{A}^b)^T \mathbf{M}^b \mathbf{J}^b - (\mathbf{J}^b)^T \mathbf{M}^b \mathbf{A}^b \mathbf{a}^b \mathbf{J}^b. \end{aligned} \tag{84}$$

The first term on the right-hand side in (84) can be simplified so that

$$\boxed{\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = -(\mathbf{J}^b)^T (\mathbf{M}^b \mathbf{A}^b \mathbf{a}^b + (\mathbf{b}^b)^T \mathbf{M}^b) \mathbf{J}^b.} \tag{85}$$

The concise expressions (82) and (85) allow for construction of the Lagrange equations in closed form. Similar expressions can be derived using the spatial and hybrid representation of twists.

For analytic investigations of the MBS dynamics, it may be useful to write the Lagrange equations in components as

$$\sum_{j=1}^n M_{ij}(\mathbf{q})\ddot{q}_j + \sum_{j,k=1}^n \Gamma_{ijk}(\mathbf{q})\dot{q}_j \dot{q}_k = Q_i(\mathbf{q}, \dot{\mathbf{q}}, t), \tag{86}$$

where the Christoffel symbols of first kind are defined as $\Gamma_{ijk} = \frac{1}{2}(\frac{\partial M_{ik}}{\partial q_j} + \frac{\partial M_{ij}}{\partial q_k} - \frac{\partial M_{jk}}{\partial q_i}) = \Gamma_{ikj}$. The recursive relations (6) give rise to the closed-form expressions

$$\Gamma_{ijk} = \frac{1}{2} \sum_{l=k}^n ((\mathbf{J}_{l,k}^b)^T \mathbf{M}_l \mathbf{ad}_{\mathbf{J}_{l,i}^b} \mathbf{J}_{l,j}^b + (\mathbf{J}_{l,j}^b)^T \mathbf{M}_l \mathbf{ad}_{\mathbf{J}_{l,i}^b} \mathbf{J}_{l,k}^b + (\mathbf{J}_{l,i}^b)^T \mathbf{M}_l \mathbf{ad}_{\mathbf{J}_{l,s}^b} \mathbf{J}_{l,r}^b) \tag{87}$$

with $i < j \leq k$ or $j \leq i < k$, $r = \max(i, j)$, $s = \min(i, j)$.

This expression for the Christoffel symbols in Lie group notation was reported in [16, 50], and already in [48] in tensor notation. This expression simplifies when Binet’s inertia tensor $\vartheta_i = \frac{1}{2} \text{tr}(\Theta_i) \mathbf{I} - \Theta_i$ is used in the mass matrix \mathbf{M}_i^b . Then (40) is replaced by $\check{\mathbf{M}}_{ic}^b = \text{diag}(\vartheta_i, m_i \mathbf{I})$, and (41) by $\check{\mathbf{M}}_i^b = \mathbf{Ad}_{\mathbf{S}_{bc}}^{-T} \check{\mathbf{M}}_{ic}^b \mathbf{Ad}_{\mathbf{S}_{bc}}^{-1}$. This leads to

$$\Gamma_{ijk} = \frac{1}{2} \sum_{l=k}^n (\mathbf{J}_{l,j}^b)^T \check{\mathbf{M}}_l \mathbf{ad}_{\mathbf{J}_{l,i}^b} \mathbf{J}_{l,i}^b \tag{88}$$

with $i < j \leq k$ or $j \leq i < k$.

Equations (87) were presented in [50, 67, 68, 70, 71], and (88) in [50]. Prior to these publications, Eqs. (87) and (88) have been reported in [48, 49] using tensor notation rather than Lie group notation. Another publication that should be mentioned is [19] where the Lagrange equations were derived using similar algebraic operations.

The above closed forms of EOM are derived using body-fixed twists. The potential benefit of using spatial or hybrid twists remains to be explored.

6 Derivatives of motion equations

In various contexts the information about the sensitivity of the MBS kinematics and dynamics is required either with respect to joint angles, geometric parameters, or dynamic parameters. Whereas it is known that the EOM of a rigid body MBS attains a form that is linear in the dynamic parameters, they depend nonlinearly on the generalized coordinates and geometry. The POE formulation provides a means to determine sensitivity with respect to kinematic parameters.

6.1 Sensitivity of motion equations

Gradients with respect to generalized coordinates are required for the linearization of the EOM (as basis for stability analysis and controller design) and for optimal control of MBS. Since the second-order and higher derivatives (13) of the body-fixed Jacobian, (21) of the spatial Jacobian, and (26) of the hybrid Jacobian are given as algebraic closed-form expressions in terms of screw products, the linearized EOM can be evaluated recursively and expressed in closed form. Using the Lie group notation, this was reported in [76].

The same results were already presented in [49] using tensor notation. Comparing the two formulations reveals once more that the matrix Lie group formulation provides a level of abstraction leading to compact expressions. A closed form for partial derivatives of the inverse mass matrix has been reported in [51], which is required for investigating the controllability of MBS. Using the body-fixed representation of twists, recursive $O(n)$ where reported in [3, 35].

6.2 Geometric sensitivity

Optimizing the design of an MBS requires information about the sensitivity with respect to geometric parameters. A recursive algorithm was reported in [35] and its parallel implementation in [3], where the partial derivatives are computed on a case-by-case basis. The Lie group formulation gives rise to a general closed-form expression. To this end, the POE formula (93) is extended as follows.

The geometry of the two bodies i and $i - 1$ connected by joint i is encoded in the constant part $\mathbf{S}_{i,i}$ and $\mathbf{S}_{i-1,i}$ in (92), respectively in \mathbf{B}_i in the formulation in (93). These are frame transformations and hence can be parameterized in terms of screw coordinates. If \mathbf{B}_i depends on $\lambda \leq 6$ geometric parameters, it is expressed as $\mathbf{B}_i(\boldsymbol{\pi}_i) = \mathbf{B}_{i0} \exp(\mathbf{U}_{i1}\pi_{i1}) \cdots \exp(\mathbf{U}_{i\lambda}\pi_{i\lambda})$. The screw coordinates \mathbf{U}_{i1} and corresponding parameters π_{i1} account for the considered variations from the nominal geometry, represented by $\mathbf{B}_{i0} \in SE(3)$. Thus the relative configuration due to joint i and the geometric variations is $\mathbf{C}_{i-1,i}(\mathbf{q}, \boldsymbol{\pi}_i) = \mathbf{B}_i(\boldsymbol{\pi}_i) \exp({}^i\mathbf{X}_i q_i)$. The key observation is that partial derivatives of $\mathbf{B}_i(\boldsymbol{\pi}_i)$ are available in closed form, as for the joint screw coordinates. Hence also the sensitivity with respect to the MBS geometry can be expressed in closed form [52]. This fact has been applied to robot calibration [21, 22], where the POE accounts for geometric imperfections to be identified.

6.3 Time derivatives of the EOM

The design of feedback-linearizing flatness-based controllers for robotic manipulators modeled as rigid body MBS actuated by elastic actuators (so-called series elastic actuators) requires the time derivatives of the inverse dynamics solution $\mathbf{Q}(t)$ [25, 66], that is, the first and second time derivatives of the EOM are necessary. Extensions of the classical recursive Newton–Euler inverse dynamics algorithms in body-fixed representations were presented in [18]. As it can be expected, the relations are very complicated. Using the presented Lie group formulation of the inverse dynamics algorithms gives rise to rather compact and thus fail-safe algorithms. This was presented in [61] for the body-fixed and hybrid version.

7 Geometric integration

This paper focuses on the MBS modeling in terms of relative (joint) coordinates. Alternatively, the MBS kinematics can be described in terms of absolute coordinates.

One of the issues addressed when modeling MBS in terms of absolute coordinates is the *kinematic reconstruction*, that is, the determination of the motion of a rigid body, represented by $\mathbf{C}(t)$, from its velocity field $\mathbf{V}(t)$. This amounts to solving one of the equations (see Appendix A2 in [60])

$$\widehat{\mathbf{V}}^b = \mathbf{C}^{-1} \dot{\mathbf{C}}, \quad \widehat{\mathbf{V}}^s = \dot{\mathbf{C}} \mathbf{C}^{-1} \quad (89)$$

together with the NE (42) or (39), respectively. Classically, the orientation is parameterized with three parameters. The problem encountered is that there is no singularity-free global parameterization of rotations with three parameters. Instead of local parameters (position and rotation angles), the absolute configurations of the rigid bodies within the MBS can be represented by $\mathbf{C}(t)$. Then a numerical integration step from time t_{k-1} to $t_k = t_{k-1} + h$ will determine the incremental configuration update $\Delta \mathbf{C}_k = \mathbf{C}_{k-1}^{-1} \mathbf{C}_k$ with $\mathbf{C}_k = \mathbf{C}(t_k)$ and $\mathbf{C}_{k-1} = \mathbf{C}(t_{k-1})$. Equations (89) are ODEs on the Lie group $SE(3)$. These can be replaced

by ODEs on the Lie algebra $se(3)$. The motion increment from t_{k-1} to t_k is parameterized as $\Delta\mathbf{C}(t) = \exp \mathbf{X}(t)$ with an algorithmic instantaneous screw coordinate vector \mathbf{X} . Then (89) are equivalent to the ODEs on the Lie algebra

$$\mathbf{V}^s = \mathbf{dexp}_X \dot{\mathbf{X}}, \quad \mathbf{V}^b = \mathbf{dexp}_{-X} \dot{\mathbf{X}}, \quad (90)$$

where $\mathbf{dexp}_X : se(3) \rightarrow se(3)$ is the right-trivialized differential of the exp mapping on $SE(3)$ [12, 58, 59, 69]. This is the basic idea of the class of Munthe-Kaas integration schemes [20, 36, 62]. This scheme has been adapted to MBS in absolute coordinates [80]. The advantage of these integration methods is that no global parameterization is necessary since the numerical integration is pursued in terms of the incremental parameters \mathbf{X} . The ODEs (90) can be solved with any vector space integration scheme (originally the Munthe-Kaas scheme uses a Runge–Kutta method) with initial value $\mathbf{X}(t_{k-1}) = \mathbf{0}$.

Recently, the geometric integration concepts were incorporated in the generalized α method [17, 41] for MBS described in absolute coordinates. In this case the representation of proper rigid body motions is crucial, as discussed in [57, 58], which is frequently incorrectly represented by $SO(3) \times \mathbb{R}^3$. Also momentum-preserving schemes were proposed [81]. It should be mentioned that the concept of geometric integration schemes on $SE(3)$ can be transferred to the kinematics of flexible bodies undergoing large deformations described as Cosserat continua. In this context the spatial description (referred to as fixed-pole formulation) has proven to be beneficial [31]. Recent results on Lie group modeling of beams can be found in [77, 78].

8 Conclusions and outlook

The computational effort of recursive $O(n)$ algorithms, but also of the formalisms for evaluating the EOM in closed form, depends on the representation of rigid body motions and of the motions of technical joints. Since the geometry of finite rigid body and relative motions is described by the Lie group $SE(3)$ and that of instantaneous motions by the screw algebra $se(3)$, Lie group theory provides the geometric framework. As already shown in [60], Lie group formulations for the MBS kinematics give rise to compact recursive formulations in terms of relative coordinates. In this paper the corresponding recursive NE algorithms were presented and related to the various $O(n)$ algorithms scattered in the literature. This allows for a comparative investigation of their efficiency in conjunction with the modeling procedure. For instance, whereas most $O(n)$ algorithms used the hybrid representation, the spatial representation, as used by Featherstone [29] and Bottasso [12] (where it is called fixed point formulation), is receiving increased attention since it gives easily rise to structure-preserving integration schemes [10–12, 31]. A conclusive investigation will be the subject of future research. Future research will also focus on combining the $O(n)$ forward dynamics algorithm by Featherstone [29], based on NE equations using spatial representations with Naudet's algorithm [64] based on Hamilton's canonical equations in hybrid representation. The use of the spatial momentum balance shall allow for momentum-preserving integration of the EOM and at the same time to reduce the number of frame transformations. A further important research topic is the derivation of structure-preserving Lie group integration schemes for which the spatial formulation of EOM will be formulation of choice.

Acknowledgements Open access funding provided by Johannes Kepler University Linz. The author acknowledges that this work has been partially supported by the Austrian COMET-K2 program of the Linz Center of Mechatronics (LCM).

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix A: Summary of basic kinematic relations

As prerequisite, the kinematic relations derived in [60] are summarized. Denote by

$$C_i = \begin{pmatrix} \mathbf{R}_i & \mathbf{r}_i \\ \mathbf{0} & 1 \end{pmatrix} \in SE(3) \tag{91}$$

the *absolute configuration* of body i with respect to the inertial frame (IFR) \mathcal{F}_0 . This is alternatively denoted by $C_i = (\mathbf{R}_i, \mathbf{r}_i)$. The *relative configuration* of body i relative to body $i - 1$ is given as

$$C_{i-1,i}(q_i) = S_{i-1,i} \exp({}^{i-1}\mathbf{Z}_i q_i) S_{i,i}^{-1} = \mathbf{B}_i \exp({}^i\mathbf{X}_i q_i), \tag{92}$$

where $\mathbf{B}_i := S_{i-1,i} S_{i,i}^{-1} = C_{i-1,i}(0)$ is the reference configuration of body i with respect to body $i - 1$, that is, for $q_i = 0$, and ${}^{i-1}\mathbf{Z}_i \in \mathbb{R}^6$ is the screw coordinate vector of joint i represented in the joint frame (JFR) $\mathcal{F}_{i-1,i}$ on body $i - 1$. Successive relative configurations can be combined to

$$C_i(\mathbf{q}) = \mathbf{B}_1 \exp({}^1\mathbf{X}_1 q_1) \cdot \mathbf{B}_2 \exp({}^2\mathbf{X}_2 q_2) \cdot \dots \cdot \mathbf{B}_i \exp({}^i\mathbf{X}_i q_i) \tag{93}$$

$$= \exp(\mathbf{Y}_1 q_1) \cdot \exp(\mathbf{Y}_2 q_2) \cdot \dots \cdot \exp(\mathbf{Y}_i q_i) \mathbf{A}_i, \tag{94}$$

where ${}^i\mathbf{X}_i \in \mathbb{R}^6$ is the screw coordinate vector of joint i represented in the joint frame fixed at body i , $\mathbf{Y}_i \in \mathbb{R}^6$ is the joint screw coordinate vector in spatial representation (measured and resolved in IFR) for the reference configuration $\mathbf{q} = \mathbf{0}$, and $\mathbf{A}_i = C_i(\mathbf{0})$ is the reference configuration of body i . The two representations of joint screw coordinates are related by

$$\mathbf{Y}_i = \mathbf{Ad}_{\mathbf{A}_i} {}^i\mathbf{X}_i, \quad {}^i\mathbf{X}_i = \mathbf{Ad}_{S_{i,i}} {}^{i-1}\mathbf{Z}_i, \tag{95}$$

where, in vector representation of screws, the adjointed transformation \mathbf{Ad} corresponding to $C \in SE(3)$ is given by the matrix

$$\mathbf{Ad}_C = \begin{pmatrix} \mathbf{R} & \mathbf{0} \\ \tilde{\mathbf{r}}\mathbf{R} & \mathbf{R} \end{pmatrix}. \tag{96}$$

For simplicity, the following notations are used:

$$\mathbf{Ad}_{\mathbf{R}} = \begin{pmatrix} \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix} \quad \text{for } C = (\mathbf{R}, \mathbf{0}), \quad \mathbf{Ad}_{\mathbf{r}} = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \tilde{\mathbf{r}} & \mathbf{I} \end{pmatrix} \quad \text{for } C = (\mathbf{I}, \mathbf{r}), \tag{97}$$

so that $\mathbf{Ad}_C = \mathbf{Ad}_{\mathbf{r}} \mathbf{Ad}_{\mathbf{R}}$.

The twist of body i in *body-fixed* representation $\mathbf{V}_i^b = (\boldsymbol{\omega}_i^b, \mathbf{v}_i^b)^T$ and in *spatial* representation $\mathbf{V}_i^s = (\boldsymbol{\omega}_i^s, \mathbf{v}_i^s)^T$ is defined by

$$\widehat{\mathbf{V}}_i^b = \begin{pmatrix} \tilde{\boldsymbol{\omega}}_i^b & \mathbf{v}_i^b \\ \mathbf{0} & 0 \end{pmatrix} = C_i^{-1} \dot{C}_i, \quad \widehat{\mathbf{V}}_i^s = \begin{pmatrix} \tilde{\boldsymbol{\omega}}_i^s & \mathbf{v}_i^s \\ \mathbf{0} & 0 \end{pmatrix} = \dot{C}_i C_i^{-1}. \tag{98}$$

Here $\mathbf{v}_i^b = \mathbf{R}_i^T \dot{\mathbf{r}}_i$ the body-fixed translational velocity, that is, the velocity of the origin of the body-fixed reference frame (RFR) \mathcal{F}_i of body i measured in the IFR \mathcal{F}_0 and resolved in \mathcal{F}_i ,

whereas $\mathbf{v}_i^s = \dot{\mathbf{r}}_i + \mathbf{r}_i \times \boldsymbol{\omega}_i^s$ is the spatial translational velocity, that is, the velocity of the point of the body that is momentarily passing through the origin of the IFR \mathcal{F}_0 resolved in the IFR. The body-fixed and spatial angular velocity, $\boldsymbol{\omega}_i^b$ and $\boldsymbol{\omega}_i^s$, are defined by $\tilde{\boldsymbol{\omega}}_i^b = \mathbf{R}_i^T \dot{\mathbf{R}}_i$ and $\tilde{\boldsymbol{\omega}}_i^s = \dot{\mathbf{R}}_i \mathbf{R}_i^T$, respectively. The *hybrid twist* is defined as $\mathbf{V}_i^h = (\boldsymbol{\omega}_i^s, \dot{\mathbf{r}}_i)^T$, and finally the *mixed twist* as $\mathbf{V}_i^m = (\boldsymbol{\omega}_i^b, \dot{\mathbf{r}}_i)^T$. The four representations are related as follows:

$$\mathbf{V}_i^h = \begin{pmatrix} \mathbf{R}_i & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_i \end{pmatrix} \mathbf{V}_i^b = \mathbf{Ad}_{\mathbf{R}_i} \mathbf{V}_i^b, \tag{99}$$

$$\mathbf{V}_i^s = \mathbf{Ad}_{\mathbf{C}_i} \mathbf{V}_i^b = \mathbf{Ad}_{\mathbf{C}_i} \mathbf{Ad}_{\mathbf{R}_i}^{-1} \mathbf{V}_i^h = \mathbf{Ad}_{\mathbf{r}_i} \mathbf{V}_i^h, \tag{100}$$

$$\mathbf{V}_i^m = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_i \end{pmatrix} \mathbf{V}_i^b = \begin{pmatrix} \mathbf{R}_i^T & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \mathbf{V}_i^h = \begin{pmatrix} \mathbf{R}_i^T & \mathbf{0} \\ -\dot{\mathbf{r}}_i & \mathbf{I} \end{pmatrix} \mathbf{V}_i^s. \tag{101}$$

The twist of body i within a kinematic chain is determined in terms of the generalized velocities $\dot{\mathbf{q}}$ as

$$\mathbf{V}_i^b = \sum_{j \leq i} \mathbf{J}_{i,j}^b \dot{q}_j = \mathbf{J}_i^b \dot{\mathbf{q}}, \quad \mathbf{V}_i^s = \sum_{j \leq i} \mathbf{J}_j^s \dot{q}_j = \mathbf{J}_i^s \dot{\mathbf{q}}, \tag{102}$$

$$\mathbf{V}_i^h = \sum_{j \leq i} \mathbf{J}_{i,j}^h \dot{q}_j = \mathbf{J}_i^h \dot{\mathbf{q}}, \quad \mathbf{V}_i^m = \sum_{j \leq i} \mathbf{J}_{i,j}^m \dot{q}_j = \mathbf{J}_i^m \dot{\mathbf{q}}, \tag{103}$$

with the Jacobian \mathbf{J}_i^b in body-fixed, \mathbf{J}_i^s in spatial, \mathbf{J}_i^h in hybrid, and \mathbf{J}_i^m in mixed representation. The i th column of the Jacobian is respectively given by

$$\mathbf{J}_{i,j}^b = \mathbf{Ad}_{\mathbf{C}_{i,j}} {}^j \mathbf{X}_j = \mathbf{Ad}_{\mathbf{C}_{i,j} \mathbf{A}_j^{-1}} \mathbf{Y}_j, \tag{104}$$

$$\mathbf{J}_j^s = \mathbf{Ad}_{\mathbf{C}_j} {}^j \mathbf{X}_j = \mathbf{Ad}_{\mathbf{C}_j \mathbf{A}_j^{-1}} \mathbf{Y}_j, \tag{105}$$

$$\mathbf{J}_{i,j}^h = \mathbf{Ad}_{\mathbf{r}_{i,j}} {}^0 \mathbf{X}_j^j \quad \text{for } j \leq i. \tag{106}$$

These are the instantaneous joint screw coordinates in body-fixed, spatial, and hybrid representation. The Jacobians are related as

$$\mathbf{J}_j^s = \mathbf{Ad}_{\mathbf{r}_i} \mathbf{J}_{i,j}^h = \mathbf{Ad}_{\mathbf{C}_i} \mathbf{J}_i^b, \quad \mathbf{J}_i^h = \mathbf{Ad}_{\mathbf{R}_i} \mathbf{J}_i^b. \tag{107}$$

The representations of joint screw coordinates are related by (95) and by

$$\mathbf{Y}_j = \mathbf{Ad}_{\mathbf{r}_i} {}^0 \mathbf{X}_j^j, \quad {}^0 \mathbf{X}_j^j = \mathbf{Ad}_{\mathbf{R}_j} {}^j \mathbf{X}_j, \tag{108}$$

where \mathbf{r}_i is the current position of body i in \mathbf{C}_i . The twists admit the recursive expressions

$$\mathbf{V}_i^b = \mathbf{Ad}_{\mathbf{C}_{i,i-1}} \mathbf{V}_{i-1}^b + {}^i \mathbf{X}_i \dot{q}_i, \tag{109}$$

$$\mathbf{V}_i^s = \mathbf{V}_{i-1}^s + \mathbf{J}_i^s \dot{q}_i, \tag{110}$$

$$\mathbf{V}_i^h = \mathbf{Ad}_{\mathbf{r}_{i,i-1}} \mathbf{V}_{i-1}^h + {}^0 \mathbf{X}_i^i \dot{q}_i. \tag{111}$$

Summarizing the twists of all bodies in $\mathbf{V}^b, \mathbf{V}^s, \mathbf{V}^h \in \mathbb{R}^{6n}$, respectively, admits the expressions

$$\mathbf{V}^b = \mathbf{J}^b \dot{\mathbf{q}}, \quad \mathbf{V}^s = \mathbf{J}^s \dot{\mathbf{q}}, \quad \mathbf{V}^h = \mathbf{J}^h \dot{\mathbf{q}} \tag{112}$$

in terms of the system Jacobians that admit the factorizations [60]

$$\mathbf{J}^b = \mathbf{A}^b \mathbf{X}^b, \quad \mathbf{J}^s = \mathbf{A}^s \mathbf{Y}^s = \mathbf{A}^{sb} \mathbf{X}^b, \quad \mathbf{J}^h = \mathbf{A}^h \mathbf{X}^h. \tag{113}$$

This provides a compact description of the overall MBS kinematics. The explicit relations for the inverse of the matrices A are the starting point for deriving recursive forward dynamics $O(n)$ algorithms.

Appendix B: Rigid body motions and the Lie group $SE(3)$

For an introduction to screws and to the motion Lie group $SE(3)$, the reader is referred to the text books [5, 47, 63, 75].

B.1 Derivatives of screws

Let C_i be time dependent. According to (96), the corresponding frame transformation of screw coordinates from \mathcal{F}_i to \mathcal{F}_0 is $\mathbf{X} \equiv {}^0\mathbf{X} = \mathbf{Ad}_{C_i} {}^i\mathbf{X}$. Assume that the screw coordinates expressed in body-fixed frame are constant. The rate of change of the screw coordinates expressed in IFR is $\frac{d}{dt} \widehat{\mathbf{X}} = \frac{d}{dt} \mathbf{Ad}_{C_i} ({}^i\widehat{\mathbf{X}}) = \dot{C}_i C_i^{-1} C_i {}^i\widehat{\mathbf{X}} C_i^{-1} - C_i {}^i\widehat{\mathbf{X}} C_i^{-1} \dot{C}_i C_i^{-1} = \widehat{\mathbf{V}}_i^s \widehat{\mathbf{X}} - \widehat{\mathbf{V}}_i^s \widehat{\mathbf{X}} = [\widehat{\mathbf{V}}_i^s, \widehat{\mathbf{X}}]$. Therein

$$[\widehat{\mathbf{X}}_1, \widehat{\mathbf{X}}_2] = \widehat{\mathbf{X}}_1 \widehat{\mathbf{X}}_2 - \widehat{\mathbf{X}}_2 \widehat{\mathbf{X}}_1 = \mathbf{ad}_{\mathbf{X}_1}(\mathbf{X}_2) \tag{114}$$

is the Lie bracket on $se(3)$, also called the *adjoint* mapping. In vector notation of screws, denoting a general screw vector with $\mathbf{X} = (\boldsymbol{\xi}, \boldsymbol{\eta})^T$, this is

$$[\mathbf{X}_1, \mathbf{X}_2] = (\boldsymbol{\xi}_1 \times \boldsymbol{\xi}_2, \boldsymbol{\eta}_1 \times \boldsymbol{\xi}_2 + \boldsymbol{\xi}_1 \times \boldsymbol{\eta}_2)^T = \mathbf{ad}_{\mathbf{X}_1} \mathbf{X}_2 \tag{115}$$

with

$$\mathbf{ad}_{\mathbf{X}} = \begin{pmatrix} \widetilde{\boldsymbol{\xi}} & \mathbf{0} \\ \widetilde{\boldsymbol{\eta}} & \widetilde{\boldsymbol{\xi}} \end{pmatrix}. \tag{116}$$

The form (115) is known as the screw product [13, 75]. Matrix (116) has appeared under different names, such as “spatial cross product” in [28, 29, 38] or the “north-east cross product” [12]. The Lie bracket obeys the Jacobi identity

$$[\mathbf{X}_1, [\mathbf{X}_2, \mathbf{X}_3]] + [\mathbf{X}_2, [\mathbf{X}_3, \mathbf{X}_1]] + [\mathbf{X}_3, [\mathbf{X}_1, \mathbf{X}_2]] = \mathbf{0}. \tag{117}$$

Allowing for time-dependent body-fixed screw coordinates ${}^i\mathbf{X}$, this relation gives rise to an expression for the time derivative of screw coordinates in moving frames:

$$\mathbf{Ad}_{C_i}^{-1} \dot{\mathbf{X}} = {}^i\dot{\mathbf{X}} + [\mathbf{v}_i^b, {}^i\mathbf{X}]. \tag{118}$$

This is the spatial extension of Euler’s formula for the derivative of a vector resolved in a moving frame.

For simplicity, throughout the paper the following notations are used:

$$\overset{\omega}{\mathbf{V}} = \begin{pmatrix} \boldsymbol{\omega} \\ \mathbf{0} \end{pmatrix}, \quad \overset{v}{\mathbf{V}} = \begin{pmatrix} \mathbf{0} \\ \mathbf{v} \end{pmatrix}. \tag{119}$$

Then the matrices

$$\mathbf{ad}_{\boldsymbol{\omega}} = \begin{pmatrix} \widetilde{\boldsymbol{\omega}} & \mathbf{0} \\ \mathbf{0} & \widetilde{\boldsymbol{\omega}} \end{pmatrix}, \quad \mathbf{ad}_{\mathbf{v}} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \widetilde{\mathbf{v}} & \mathbf{0} \end{pmatrix} \tag{120}$$

are used to denote matrix (116) for twists $\overset{\omega}{\mathbf{V}}$ and $\overset{v}{\mathbf{V}}$, respectively, which are the infinitesimal versions on (97).

B.2 Wrenches as coscrews— $se^*(3)$

Screws are the geometric objects embodying twists, wrenches, and momenta of rigid bodies. These different physical meanings imply different mathematical interpretations of the geometric object.

A wrench, defined by a force and moment, is denoted by $\mathbf{W} = (\mathbf{t}, \mathbf{f})^T$. The force applied at a point with position vector \mathbf{p} generates the moment $\mathbf{t} = \mathbf{p} \times \mathbf{f}$. The dual to the Chasles theorem is the Poincot theorem stating that every system of forces can be reduced to a force together with a couple with moment parallel to the force.

Geometrically, a screw is determined by the Plücker coordinates of the line along the screw axis and the pitch. If \mathbf{e} is the unit vector along the screw axis, and \mathbf{p} is a position vector of a point on that axis, then the screw coordinate vector of a twist is $\mathbf{V} = (\boldsymbol{\omega}, \mathbf{v})^T = \boldsymbol{\omega}(\mathbf{e}, \mathbf{p} \times \mathbf{e} + h\mathbf{e})^T$, where $\omega = \|\boldsymbol{\omega}\|$ is its magnitude, and $h = \mathbf{v}^T \boldsymbol{\omega} / \omega^2$ is its pitch. The screw coordinate vector of a wrench, that is, the force \mathbf{f} producing a torque \mathbf{t} about the axis \mathbf{e} when the point of application is displaced according to \mathbf{p} from the axis, is $\mathbf{W} = (\mathbf{t}, \mathbf{f})^T = \mathbf{f}(\mathbf{p} \times \mathbf{e} + h\mathbf{e}, \mathbf{e})^T$ with pitch $h = \mathbf{t}^T \mathbf{f} / \|\mathbf{f}\|^2$. Apparently, the linear and angular components of the screw coordinates are interchanged for twists and wrenches. The different definition of screw coordinate vectors allows to describe the action of a wrench on a twist as the scalar product: $\mathbf{W}^T \mathbf{V}$ is the power performed by the wrench acting on twist \mathbf{V} .

A twist ${}^2\mathbf{V}$ represented in frame \mathcal{F}_2 transforms to its representation in frame \mathcal{F}_1 according to ${}^1\mathbf{V} = \text{Ad}_{\mathbf{S}_{1,2}} {}^2\mathbf{V}$. The power conservation yields that a wrench represented in \mathcal{F}_1 transforms to its representation in \mathcal{F}_2 according to

$${}^2\mathbf{W} = \text{Ad}_{\mathbf{S}_{1,2}}^T {}^1\mathbf{W}. \tag{121}$$

Although this notation is useful for kinetostatic formulations, it is inconsistent in the sense that it treats screw coordinates differently for twists and wrenches. In screw theory, aiming on a consistent treatment of screw entities, a screw is represented by its coordinates as defined by (67) in [60], and the so-called *reciprocal product* of two screws is used [5, 13, 75]. The latter is defined for $\mathbf{X}_1 = (\boldsymbol{\xi}_1, \boldsymbol{\eta}_1)^T$ and $\mathbf{X}_2 = (\boldsymbol{\xi}_2, \boldsymbol{\eta}_2)^T$ as $\mathbf{X}_1 \odot \mathbf{X}_2 = \boldsymbol{\xi}_1^T \boldsymbol{\eta}_2 + \boldsymbol{\eta}_1^T \boldsymbol{\xi}_2$. Two screws are said to be *reciprocal* if $\mathbf{X}_1 \odot \mathbf{X}_2 = 0$. Obviously, if twists and wrenches are represented consistently with the same definition of screw coordinates, then a twist and wrench screws being reciprocal means that they perform no work. Geometrically, for zero pitch screws, this means that the screw axes intersect.

In screw theory, wrench screws are called coscrews to distinguish them from motion screws and to indicate that a wrench acts on a motion screw (a twist) as a linear operator that returns work or power. Since twists form the Lie algebra $se(3)$, wrenches form the dual $se^*(3)$.

Appendix C: Nomenclature

\mathcal{F}_0	IFR
\mathcal{F}_i	BFR of body i
$\mathcal{J}_{i,i}$	JFR for joint i at body i , joint i connects body i with its predecessor body $i - 1$
$\mathcal{J}_{i-1,i}$	JFR for joint i at body $i - 1$
${}^i\mathbf{r}$	Coordinate representation of a vector resolved in BFR on body i . The index is omitted if this is the IFR: $\mathbf{r} \equiv {}^0\mathbf{r}$

\mathbf{R}_i	Rotation matrix from BFR \mathcal{F}_i at body i to IFR \mathcal{F}_0
$\mathbf{R}_{i,j}$	Rotation matrix transforming coordinates resolved in BFR \mathcal{F}_j to coordinates resolved in \mathcal{F}_i
\mathbf{r}_i	Position vector of origin of BFR \mathcal{F}_i at body i resolved in IFR \mathcal{F}_0
$\mathbf{r}_{i,j}$	Position vector from origin of BFR \mathcal{F}_i to origin of BFR \mathcal{F}_j
$\tilde{\mathbf{X}}$	skew symmetric matrix associated with the vector $\mathbf{x} \in \mathbb{R}^3$
$\mathbf{C}_i = (\mathbf{R}_i, \mathbf{r}_i)$	Absolute configuration of body i . This is denoted in matrix form by \mathbf{C}_i
$\mathbf{C}_{i,j} = \mathbf{C}_i^{-1}\mathbf{C}_j$	Relative configuration of body j w.r.t. body i
${}^k\mathbf{v}_i^j$	Translational velocity of body i measured at origin of BFR \mathcal{F}_j , resolved in BFR \mathcal{F}_k
$\mathbf{v}_i^b \equiv {}^i\mathbf{v}_i^i$	Body-fixed representation of the translational velocity of body i
$\mathbf{v}_i^s \equiv {}^0\mathbf{v}_i^0$	Spatial representation of the translational velocity of body i
${}^k\boldsymbol{\omega}_i$	Angular velocity of body i measured and resolved in BFR \mathcal{F}_k
$\boldsymbol{\omega}_i^b \equiv {}^i\boldsymbol{\omega}_i$	Body-fixed representation of the angular velocity of body i
$\boldsymbol{\omega}_i^s \equiv {}^0\boldsymbol{\omega}_i$	Spatial representation of the angular velocity of body i
${}^k\mathbf{V}_i^j$	Twist (RFR of) body i measured in \mathcal{F}_j and resolved in \mathcal{F}_k
$\mathbf{V}_i^b \equiv {}^i\mathbf{V}_i^i$	Body-fixed representation of the twist of body i
$\mathbf{V}_i^s \equiv {}^0\mathbf{V}_i^0$	Spatial representation of the twist of body i
$\mathbf{V}_i^h \equiv {}^0\mathbf{V}_i^i$	Hybrid form of the twist of body i
\mathbf{V}^b	Vector of system twists in body-fixed representation
\mathbf{V}^s	Vector of system twists in spatial representation
\mathbf{V}^h	Vector of system twists in hybrid representation
\mathbf{V}^m	Vector of system twists in mixed representation
\mathbf{W}_i^b	Applied wrench at body i in body-fixed representation
\mathbf{W}_i^s	Applied wrench at body i in spatial representation
\mathbf{W}_i^h	Applied wrench at body i in hybrid representation
\mathbf{M}_i^b	Inertia matrix of body i in body-fixed representation
\mathbf{M}_i^s	Inertia matrix of body i in spatial representation
\mathbf{M}_i^h	Inertia matrix of body i in hybrid representation
\mathbf{Ad}_R	Screw transformation associated with $C = (\mathbf{R}, \mathbf{0})$
\mathbf{Ad}_r	Screw transformation associated with $C = (\mathbf{I}, \mathbf{r})$
$\mathbf{Ad}_{C_{i,j}}$	Transformation matrix transforming screw coordinates represented in \mathcal{F}_j to screw coordinates represented in \mathcal{F}_i
\mathbf{ad}_X	Screw product matrix associated with screw coordinate vector $\mathbf{X} \in \mathbb{R}^6$
$[\mathbf{X}, \mathbf{Y}]$	Lie bracket of screw coordinate vectors $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^6$; $[\mathbf{X}, \mathbf{Y}] = \mathbf{ad}_X\mathbf{Y}$
$\widehat{\mathbf{X}} \in se(3)$	4×4 matrix associated with the screw coordinate vectors $\mathbf{X} \in \mathbb{R}^6$
$SE(3)$	Special Euclidean group in three dimensions—Lie group of rigid body motions
$se(3)$	Lie algebra of $SE(3)$, algebra of screws
$\mathbf{q} \in \mathbb{V}^n$	Joint coordinate vector
\mathbb{V}^n	Configuration space

References

1. Anderson, K.: An order n formulation for the motion simulation of general multi-rigid-body constrained systems. *Comput. Struct.* **43**(3), 565–579 (1992)
2. Anderson, K.S., Critchley, J.H.: Improved ‘order- N ’ performance algorithm for the simulation of constrained multi-rigid-body dynamic systems. *Multibody Syst. Dyn.* **9**(2), 185–212 (2003)

3. Anderson, K.S., Hsu, Y.: Order- $(n + m)$ direct differentiation determination of design sensitivity for constrained multibody dynamic systems. *Struct. Multidiscip. Optim.* **26**, 171–182 (2004)
4. Angeles, J., Lee, S.: The formulation of dynamical equations of holonomic mechanical systems using a natural orthogonal complement. *J. Appl. Mech.* **55**, 243–244 (1988)
5. Angeles, J.: *Fundamentals of Robotic Mechanical Systems*, 2nd edn. Springer, Berlin (2003)
6. Bae, D.S., Hwang, R.S., Haug, E.J.: A recursive formulation for real-time dynamic simulation of mechanical systems. *J. Mech. Des.* **113**(2), 158–166 (1988)
7. Bae, D.S., et al.: A generalized recursive formulation for constrained flexible multibody dynamics. *Int. J. Numer. Methods Eng.* **50**, 1841–1859 (2001)
8. Benedict, C.E., Tesar, D.: Model formulation of complex mechanisms with multiple inputs: part I—geometry. *J. Mech. Des.* **100**(4), 747–754 (1978)
9. Bongardt, B., Kirchner, F.: Newton–Euler equations in general coordinates. In: *Proc. of Mathematics of Robotics (IMA)*, September 9–11, St Anne’s College, University of Oxford (2015)
10. Borri, M., Bottasso, C.L., Trainelli, L.: Integration of elastic multibody systems by invariant conserving/dissipating algorithms—part II: numerical schemes and applications. *Comput. Methods Appl. Mech. Eng.* **190**(29/30), 3701–3733 (2001)
11. Borri, M., Bottasso, C.L., Trainelli, L.: An invariant-preserving approach to robust finite-element multibody simulation. *Z. Angew. Math. Mech.* **83**(10), 663–676 (2003)
12. Bottasso, C.L., Borri, M.: Integrating finite rotations. *Comput. Methods Appl. Mech. Eng.* **164**, 307–331 (1998)
13. Bottema, O., Roth, B.: *Theoretical Kinematics*. Dover, New York (1990)
14. Bremer, H.: *Elastic Multibody Dynamics*. Springer, Berlin (2010)
15. Brockett, R.W.: Robotic manipulators and the product of exponentials formula. In: *Mathematical Theory of Networks and Systems. Lecture Notes in Control and Information Sciences*, vol. 58, pp. 120–129 (1984)
16. Brockett, R.W., Stokes, A., Park, F.: A geometrical formulation of the dynamical equations describing kinematic chains. *IEEE Int. Conf. Robot. Autom.* **2**, 637–641 (1993)
17. Brüls, O., Cardona, A., Arnold, M.: Lie group generalized-alpha time integration of constrained flexible multibody systems. *Mech. Mach. Theory* **48**, 121–137 (2012)
18. Buondonno, G., De Luca, A.: A recursive Newton–Euler algorithm for robots with elastic joints and its application to control. 2015 *IEEE/RSJ IROS*, 5526–5532
19. Burdick, J.W.: An algorithm for generation of efficient manipulator dynamic equations. In: *Proc. IEEE Int. Conf. Rob. and Automat. (ICRA)*, vol. 3, pp. 212–218 (1986)
20. Celledoni, E., Owren, B.: Lie group methods for rigid body dynamics and time integration on manifolds. *Comput. Methods Appl. Mech. Eng.* **19**, 421–438 (1999)
21. Chen, I.-M., Yang, G.: Kinematic calibration of modular reconfigurable robots using product-of-exponentials formula. *J. Robot. Syst.* **14**(11), 807–821 (1997)
22. Chen, I.-M., et al.: Local POE model for robot kinematic calibration. *Mech. Mach. Theory* **36**(11–12), 1215–1239 (2001)
23. Chevallier, D.P.: Lie algebras, modules, dual quaternions and algebraic methods in kinematics. *Mech. Mach. Theory* **26**(6), 613–627 (1991)
24. Chevallier, D.P.: Lie groups and multibody dynamics formalism. In: *Proc. EUROMECH Colloquium, Prague*, vol. 320, pp. 1–20 (1994)
25. De Luca, A.: Decoupling and feedback linearization of robots with mixed rigid/elastic joints. *Int. J. Robust Nonlinear Control* **8**, 965–977 (1998)
26. Featherstone, R.: The calculation of robot dynamics using articulated-body inertias. *Int. J. Robot. Res.* **2**(1), 13–30 (1983)
27. Featherstone, R.: *Robot Dynamics Algorithms*. Kluwer Academic, Dordrecht (1987)
28. Featherstone, R.: The acceleration vector of a rigid body. *Int. J. Robot. Res.* **20**, 841–846 (2001)
29. Featherstone, R.: *Rigid Body Dynamics Algorithms*. Springer, Berlin (2008)
30. Fijany, A.: Parallel $O(\log N)$ algorithms for computation of manipulator forward dynamics. *IEEE Trans. Robot. Autom.* **11**(3), 389–400 (1995)
31. Gačša, M., Jelenić, G.: Modified fixed-pole approach in geometrically exact spatial beam finite elements. *Finite Elem. Anal. Des.* **99**, 39–48 (2015)
32. Hervé, J.M.: Analyse Structurelle des Mécanismes par Groupe des Déplacements. *Mech. Mach. Theory* **13**, 437–450 (1978)
33. Hervé, J.M.: Intrinsic formulation of problems of geometry and kinematics of mechanisms. *Mech. Mach. Theory* **17**(3), 179–184 (1982)
34. Hollerbach, J.M.: A recursive Lagrangian formulation of manipulator dynamics and a comparative study of dynamics formulation complexity. *IEEE Trans. Syst. Man Cybern.* **SMC-10**(11), 730–736 (1980)

35. Hsu, Y., Anderson, K.S.: Recursive sensitivity analysis for constrained multi-rigid-body dynamic systems design optimization. *Struct. Multidiscip. Optim.* **24**, 312–324 (2002)
36. Iserles, A., et al.: Lie-group methods. *Acta Numer.* **9**, 215–365 (2000)
37. Jain, A.: Unified formulation of dynamics for serial rigid multibody systems. *J. Guid. Control Dyn.* **14**(3), 531–542 (1991)
38. Jain, A., Rodriguez, G.: Diagonalized Lagrangian robot dynamics. *IEEE Trans. Robot. Autom.* **11**(4), 571–584 (1995)
39. Jungnickel, U., Maisser, P.: Dynamiksimulation von Mehrkörpersystemen mit geschlossenensn kinematischen Ketten mittels Algebrodifferentialgleichungen. Report No. 26, Institut für Mechanik, Akademie der Wissenschaften der DDR (1990). ISSN 0232-5330
40. Kane, T.R., Levinson, D.A.: *Dynamics: Theory and Applications*. Mcgraw Hill, New York (1985)
41. Krysl, P., Endres, L.: Explicit Newmark/Verlet algorithm for time integration of the rotational dynamics of rigid bodies. *Int. J. Numer. Methods Eng.* **62**, 2154–2177 (2005)
42. Legnani, G.: A homogeneous matrix approach to 3D kinematics and dynamics—I. *Theory. Mech. Mach. Theory* **31**(5), 573–587 (1996)
43. Legnani, G.: A homogeneous matrix approach to 3D kinematics and dynamics—II. Applications to chains of rigid bodies and serial manipulators. *Mech. Mach. Theory* **31**(5), 589–605 (1996)
44. Lilly, W., Orin, D.E.: Alternate formulations for the manipulator inertia matrix. *Int. J. Robot. Res.* **10**(1), 64–74 (1991)
45. Liu, Y.: Screw-matrix method in dynamics of multibody systems. *Acta Mech. Sin.* **4**(2), 165–174 (1988)
46. Luh, J.Y.S., Walker, M.W., Paul, R.P.C.: On-line computational scheme for mechanical manipulators. *J. Dyn. Syst. Meas. Control* **102**(2), 69–76 (1980)
47. Lynch, K.M., Park, F.C.: *Modern Robotics*. Cambridge University Press, Cambridge (2017)
48. Maißer, P.: Analytische Dynamik von Mehrkörpersystemen. *Z. Angew. Math. Mech.* **68**(10), 463–481 (1988)
49. Maisser, P.: Differential-geometric methods in multibody dynamics. *Nonlinear Anal.* **30**(8), 5127–5133 (1997)
50. Müller, A., Maisser, P.: Lie group formulation of kinematics and dynamics of constrained MBS and its application to analytical mechanics. *Multibody Syst. Dyn.* **9**, 311–352 (2003)
51. Müller, A.: Partial derivatives of the inverse mass matrix of multibody systems via its factorization. *IEEE Trans. Robot.* **23**(1), 164–168 (2007)
52. Müller, A.: Generic mobility of rigid body mechanisms. *Mech. Mach. Theory* **44**(6), 1240–1255 (2009)
53. Müller, A.: MBS motion equation without explicit definition of body-fixed reference frames, 38 mechanisms and robotics conference (MECH). ASME 2014 International Design Engineering Technical Conferences, August 12–15, 2014, Buffalo, NY
54. Müller, A.: Higher derivatives of the kinematic mapping and some applications. *Mech. Mach. Theory* **76**, 70–85 (2014)
55. Müller, A.: Derivatives of screw systems in body-fixed representation. In: Lenarcic, J., Khatib, O. (eds.) *Advances in Robot Kinematics (ARK)*. Springer, Berlin (2014)
56. Müller, A.: Recursive higher-order constraints for linkages with lower kinematic pairs. *Mech. Mach. Theory* **100**, 33–43 (2016)
57. Müller, A., Terze, Z.: The significance of the configuration space Lie group for the constraint satisfaction in numerical time integration of multibody systems. *Mech. Mach. Theory* **82**, 173–202 (2014)
58. Müller, A.: A note on the motion representation and configuration update in time stepping schemes for the constrained rigid body. *BIT Numer. Math.* **56**(3), 995–1015 (2016)
59. Müller, A.: Coordinate mappings for rigid body motions. *J. Comput. Nonlinear Dyn.* **12**(2), 021010 (2016)
60. Müller, A.: Screw and Lie group theory in multibody kinematics—motion representation and recursive kinematics of tree-topology systems. *Multibody Syst. Dyn.* (2017). doi:10.1007/s11044-017-9582-7
61. Müller, A.: Recursive second-order inverse dynamics for serial manipulators. In: *IEEE Int. Conf. Robotics Automations (ICRA)*, May 29–June 3, 2017, Singapore
62. Munthe-Kaas, H.: High order Runge–Kutta methods on manifolds. *Appl. Numer. Math.* **29**, 115–127 (1999)
63. Murray, R.M., Li, Z., Sastry, S.S.: *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton (1994)
64. Naudet, J., Lefeber, D., Daerden, F., Terze, Z.: Forward dynamics of open-loop multibody mechanisms using an efficient recursive algorithm based on canonical momenta. *Multibody Dyn.* **10**(1), 45–59 (2003)
65. Orin, D., Schrader, W.: Efficient computation of the Jacobian for robot manipulators. *Int. J. Robot. Res.* **3**(4) (1984)
66. Palli, G., Melchiorri, C., De Luca, A.: On the feedback linearization of robots with variable joint stiffness. In: *IEEE Int. Conf. Rob. Aut. (IROS)*, Pasadena, CA, USA May 19–23 (2008)

67. Park, F.C.: Computational aspects of the product-of-exponentials formula for robot kinematics. *IEEE Trans. Autom. Control* **39**(3), 643–647 (1994)
68. Park, F.C., Bobrow, J.E., Ploen, S.R.: A Lie group formulation of robot dynamics. *Int. J. Robot. Res.* **14**(6), 609–618 (1995)
69. Park, J., Chung, W.K.: Geometric integration on Euclidean group with application to articulated multi-body systems. *IEEE Trans. Robot. Autom.* **21**(5), 850–863 (2005)
70. Park, F.C., Kim, M.W.: Lie theory, Riemannian geometry, and the dynamics of coupled rigid bodies. *Z. Angew. Math. Phys.* **51**, 820–834 (2000)
71. Ploen, S.R., Park, F.C.: A Lie group formulation of the dynamics of cooperating robot systems. *Robot. Auton. Syst.* **21**, 279–287 (1997)
72. Rodriguez, G.: Kalman filtering, smoothing, and recursive robot arm forward and inverse dynamics. *IEEE J. Robot. Autom.* **RA-3**(6), 624–639 (1987)
73. Rodriguez, G., Jain, A., Kreutz-Delgado, K.: A spatial operator algebra for manipulator modelling and control. *Int. J. Robot. Res.* **10**(4), 371–381 (1991)
74. Rodriguez, G., Jain, A., Kreutz-Delgado, K.: Spatial operator algebra for multibody system dynamics. *J. Astronaut. Sci.* **40**, 27–50 (1992)
75. Selig, J.: *Geometric Fundamentals of Robotics*. Monographs in Computer Science Series. Springer, New York (2005)
76. Sohl, G.A., Bobrow, J.E.: A recursive multibody dynamics and sensitivity algorithm for branched kinematic chains. *J. Dyn. Syst. Meas. Control* **123**, 391–399 (2001)
77. Sonnevile, V., Cardona, A., Brüls, O.: Geometrically exact beam finite element formulated on the special Euclidean group SE(3). *Comput. Methods Appl. Mech. Eng.* **268**(1), 451–474 (2014)
78. Sonnevile, V., Brüls, O.: A formulation on the special Euclidean group for dynamic analysis of multi-body systems. *J. Comput. Nonlinear Dyn.* **9**(4), 041002 (2014)
79. Stelzle, W., Kecskeméthy, A., Hiller, M.: A comparative study of recursive methods. *Arch. Appl. Mech.* **66**, 9–19 (1995)
80. Terze, Z., Müller, A., Zlatar, D.: Lie-group integration method for constrained multibody systems in state space. *Multibody Syst. Dyn.* **34**(3), 275–305 (2015). doi:[10.1007/s11044-014-9439-2](https://doi.org/10.1007/s11044-014-9439-2)
81. Terze, Z., Müller, A., Zlatar, D.: An angular momentum and energy conserving Lie-group integration scheme for rigid body rotational dynamics originating from Störmer–Verlet algorithm. *J. Comput. Nonlinear Dyn.* **10**(5), 051005 (2015). 11 pages. doi:[10.1115/1.4028671](https://doi.org/10.1115/1.4028671)
82. Thomas, M., Tesar, D.: Dynamic modeling of serial manipulator arms. *J. Dyn. Syst. Meas. Control* **104**(3), 218–228 (1982)
83. Uicker, J.J., Ravani, B., Sheth, P.N.: *Matrix Methods in the Design Analysis of Mechanisms and Multi-body Systems*. Cambridge University Press, Cambridge (2013)
84. Wittenburg, J.: *Dynamics of Systems of Rigid Bodies*. B. G. Teubner, Stuttgart (1977)
85. Yamane, K., Nakamura, Y.: Comparative study on serial and parallel forward dynamics algorithms for kinematic chains. *Int. J. Robot. Res.* **28**, 622–629 (2009)