Efficient Computation of Higher-Order Variational Integrators in Robotic Simulation and Trajectory Optimization

Taosha Fan Jarvis Schultz Todd Murphey

Department of Mechanical Engineering, Northwestern University,
2145 Sheridan Road, Evanston, IL 60208, USA
taosha.fan@u.northwestern.edu, jschultz@northwestern.edu,
t-murphey@northwestern.edu

Abstract. This paper addresses the problem of efficiently computing higher-order variational integrators in simulation and trajectory optimization of mechanical systems as those often found in robotic applications. We develop $O(n)$ algorithms to evaluate the discrete Euler-Lagrange (DEL) equations and compute the Newton direction for solving the DEL equations, which results in linear-time variational integrators of arbitrarily high order. To our knowledge, no linear-time higher-order variational or even implicit integrators have been developed before. Moreover, an $O(n^2)$ algorithm to linearize the DEL equations is presented, which is useful for trajectory optimization. These proposed algorithms eliminate the bottleneck of implementing higher-order variational integrators in simulation and trajectory optimization of complex robotic systems. The efficacy of this paper is validated through comparison with existing methods, and implementation on various robotic systems—including trajectory optimization of the Spring Flamingo robot, the LittleDog robot and the Atlas robot. The results illustrate that the same integrator can be used for simulation and trajectory optimization in robotics, preserving mechanical properties while achieving good scalability and accuracy.

1 Introduction

Variational integrators conserve symplectic form, constraints and energetic quantities [1–6]. As a result, variational integrators generally outperform the other types of integrators with respect to numerical accuracy and stability, thus permitting large time steps in simulation and trajectory optimization, which is useful for complex robotic systems [1–6]. Moreover, variational integrators can also be regularized for collisions and friction by leveraging the linear complementarity problem (LCP) formulation [7, 8].

The computation of variational integrators is comprised of the discrete Euler-Lagrange equation (DEL) evaluation, the descent direction computation for solving the DEL equations and the DEL equation linearization. The computation of these three phases of variational integrators can be accomplished with automatic differentiation and our

This material is based upon work supported by the National Science Foundation under award DCSD-1662233. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

prior methods [2,4], both of which are $O(n^2)$ to evaluate the DEL equations and $O(n^3)$ to compute the Newton direction and linearize the DEL equations for an n-degree-of-freedom mechanical system. Recently, a linear-time second-order variational integrator was developed in [9], which uses the quasi-Newton method and works for small time steps and comparatively simple mechanical systems.

Higher-order variational integrators are needed for greater accuracy in predicting the dynamic motion of robots [10, 11]. However, the computation of higher-order variational integrators has rarely been addressed. The quasi-Newton method in [9] only applies to second-order variational integrators, and while automatic differentiation and our prior methods [2, 4] are implementable for higher-order variational integrators, the complexity increases superlinearly as the integrator order increases.

In this paper, we address the computation efficiency of higher-order variational integrators and develop: i) an $O(n)$ method for the evaluation of the DEL equations, ii) an $O(n)$ method for the computation of the Newton direction, and iii) an $O(n^2)$ method for the linearization of the DEL equations. The proposed characteristics i) – iii) eliminate the bottleneck of implementing higher-order variational integrators in simulation and trajectory optimization of complex robotic systems, and to the best of our knowledge, no similar work has been presented before. In particular, we believe that the resulting variational integrator from i) and ii) is the first exactly linear-time implicit integrator of third or higher order for mechanical systems.

The rest of this paper is organized as follows. Section 2 reviews higher-order variational integrators, the Lie group formulation of rigid body motion and the tree representation of mechanical systems. Sections 3 and 4 respectively detail the linear-time higher-order variational integrator and the quadratic-time linearization, which are the main contributions of this paper. Section 5 compares our work with existing methods, and Section 6 presents examples of trajectory optimization for the Spring Flamingo robot, the LittleDog robot and the Atlas robot. The conclusions are made in Section 7.

2 Preliminaries and Notation

In this section, we review higher-order variational integrators, the Lie group formulation of rigid body motion, and the tree representation of mechanical systems. In addition, notation used throughout this paper is introduced accordingly.

2.1 Higher-Order Variational Integrators

In this paper, higher-order variational integrators are derived with the methods in [1, 12, 13].

A trajectory $(q(t),\, \dot{q}(t))$ where $0 \leq t \leq T$ of a forced mechanical system should satisfy the *Lagrange-d'Alembert principle*:

$$\delta\mathfrak{S} = \delta \int_0^T \mathcal{L}(q, \dot{q})dt + \int_0^T \mathcal{F}(t) \cdot \delta q dt = 0 \tag{1}$$

in which $\mathcal{L}(q, \dot{q})$ is the system's Lagrangian and $\mathcal{F}(t)$ is the generalized force. Provided that the time interval $[0,\, T]$ is evenly divided into N sub-intervals with $\Delta t = T/N$, and each $q(t)$ over $[k\Delta t,\ (k+1)\Delta t]$ is interpolated with $s+1$ control points $q^{k,\alpha} = q(t^{k,\alpha})$ in which $\alpha = 0,\ 1,\ \cdots,\ s$ and $k\Delta t = t^{k,0} < t^{k,1} < \cdots < t^{k,s} = (k+1)\Delta t$, then there are coefficients $b^{\alpha\beta}$ ($0 \leq \alpha,\ \beta \leq s$) such that

$$\dot{q}(t^{k,\alpha}) \approx \dot{q}^{k,\alpha} = \frac{1}{\Delta t}\sum_{\beta=0}^{s} b^{\alpha\beta} q^{k,\beta}. \tag{2}$$

In this paper, we assume that the quadrature points of the quadrature rule are also $t^{k,\alpha}$ though our algorithms in Sections 3 and 4 can be generalized for any quadrature rules. Then the Lagrange-d'Alembert principle Eq. (1) is approximated as

$$\delta\mathfrak{S} \approx \sum_{k=0}^{N-1}\sum_{\alpha=0}^{s} w^{\alpha}\left[\delta\mathcal{L}(q^{k,\alpha},\dot{q}^{k,\alpha}) + \mathcal{F}(t^{k,\alpha})\cdot\delta q^{k,\alpha}\right]\cdot\Delta t = 0, \tag{3}$$

in which w^{α} are weights of the quadrature rule used for integration. In variational integrators, the *discrete Lagrangian* and the *discrete generalized force* are defined to be

$$\mathcal{L}_d(q^{k,0},\ q^{k,1},\ \cdots,\ q^{k,s}) = \sum_{\alpha=0}^{s} w^{\alpha}\mathcal{L}(q^{k,\alpha},\dot{q}^{k,\alpha})\Delta t \tag{4}$$

and $\mathcal{F}_d^{k,\alpha}(t^{k,\alpha}) = w^{\alpha}\mathcal{F}(t^{k,\alpha})\Delta t$, respectively. Note that by definition we have $t^{k,s} = t^{k+1,0}$ and $q^{k,s} = q^{k+1,0}$, and as a result of Eq. (3), we obtain

$$p^k + \mathbb{D}_1\mathcal{L}_d(\overline{q}^k) + \mathcal{F}_d^{k,0} = 0, \tag{5a}$$

$$\mathbb{D}_{\alpha+1}\mathcal{L}_d(\overline{q}^k) + \mathcal{F}_d^{k,\alpha} = 0 \quad \forall\alpha = 1,\ \cdots,\ s-1, \tag{5b}$$

$$p^{k+1} = \mathbb{D}_{s+1}\mathcal{L}_d(\overline{q}^k) + \mathcal{F}_d^{k,s} \tag{5c}$$

in which p^k is the *discrete momentum*, $\overline{q}^k$ stands for the tuple $(q^{k,0},\ q^{k,1},\ \cdots,\ q^{k,\alpha})$, and $\mathbb{D}_{\alpha+1}\mathcal{L}_d$ is the derivative with respect to $q^{k,\alpha}$. Note that Eq. (5) is known as the *discrete Euler-Lagrangian (DEL) equations*, which implicitly define an update rule $(q^{k,0},\ p^k) \to (q^{k+1,0},\ p^{k+1})$ by solving sn nonlinear equations from Eqs. (5a) and (5b). In a similar way, for mechanical systems with constraints $h(q,\dot{q}) = 0$, we have

$$p^k + \mathbb{D}_1\mathcal{L}_d(\overline{q}^k) + \mathcal{F}_d^{k,0} + A^{k,0}(q^{k,0})\cdot\lambda^{k,0} = 0, \tag{6a}$$

$$\mathbb{D}_{\alpha+1}\mathcal{L}_d(\overline{q}^k) + \mathcal{F}_d^{k,\alpha} + A^{k,\alpha}(q^{k,\alpha})\cdot\lambda^{k,\alpha} = 0 \quad \forall\alpha = 1,\ \cdots,\ s-1, \tag{6b}$$

$$p^{k+1} = \mathbb{D}_{s+1}\mathcal{L}_d(\overline{q}^k) + \mathcal{F}_d^{k,s}, \tag{6c}$$

$$h^{k,\alpha}(q^{k+1,\alpha},\dot{q}^{k+1,\alpha}) = 0 \quad \forall\alpha = 1,\ \cdots,\ s \tag{6d}$$

in which $A^{k,\alpha}(q^{k,\alpha})$ is the discrete constraint force matrix and $\lambda^{k,\alpha}$ is the discrete constraint force.

The resulting higher-order variational integrator is referred as the Galerkin integrator [1, 12, 13], the accuracy of which depends on the number of control points as well as the numerical quadrature of the discrete Lagrangian. If there are $s+1$ control points and the Lobatto quadrature is employed, then the resulting variational integrator has an accuracy of order $2s$ [12, 13]. The Galerkin integrator includes the *trapezoidal variational integrator* and the *Simpson variational integrator* as shown in Examples 1 and 2, the DEL equations of which are given by Eqs. (5) and (6).

Example 1. The trapezoidal variational integrator is a second-order integrator with two control points $\overline{q}^k = (q^{k,0},\, q^{k,1})$ such that $q^{k,0} = q\big(k\Delta t\big)$ and $q^{k,1} = q\big((k+1)\Delta t\big)$, $\dot{q}^{k,0} = \dot{q}^{k,1} = \frac{q^{k,1}-q^{k,0}}{\Delta t}$, and $\mathcal{L}_d(\overline{q}^k) = \frac{\Delta t}{2}\left[\mathcal{L}(q^{k,0},\dot{q}^{k,0}) + \mathcal{L}(q^{k,1},\dot{q}^{k,1})\right]$.

Example 2. The Simpson variational integrator is a fourth-order integrator with three control points $\overline{q}^k = (q^{k,0},\, q^{k,1},\, q^{k,2})$ in which $q^{k,0} = q\big(k\Delta t\big)$, $q^{k,0} = q\big((k+\frac{1}{2})\Delta t\big)$ and $q^{k,2} = q\big((k+1)\Delta t\big)$, $\dot{q}^{k,0} = \frac{4q^{k,1}-3q^{k,0}-q^{k,2}}{\Delta t}$, $\dot{q}^{k,1} = \frac{q^{k,2}-q^{k,0}}{\Delta t}$ and $\dot{q}^{k,2} = \frac{q^{k,0}+3q^{k,2}-4q^{k,1}}{\Delta t}$, and $\mathcal{L}_d(\overline{q}^k) = \frac{\Delta t}{6}\big[\mathcal{L}(q^{k,0},\dot{q}^{k,0}) + 4\mathcal{L}(q^{k,1},\dot{q}^{k,1}) + \mathcal{L}(q^{k,2},\dot{q}^{k,2})\big]$.

2.2 The Lie Group Formulation of Rigid Body Motion

The configuration of a rigid body $g = (R,p) \in SE(3)$ can be represented as a 4×4 matrix $g = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}$ in which $R \in SO(3)$ is a rotation matrix and $p \in \mathbb{R}^3$ is a position vector. The body velocity of the rigid body $v = (\omega, v_O) \in T_eSE(3)$ is an element of the Lie algebra and can be represented either as a 6×1 vector $v = (g^{-1}\dot{g})^\vee = \begin{bmatrix}\omega^T & v_O^T\end{bmatrix}$ or a 4×4 matrix $\hat{v} = g^{-1}\dot{g} = \begin{bmatrix} \hat{\omega} & v_O \\ 0 & 0 \end{bmatrix}$ in which $\omega = (\omega_x,\omega_y,\omega_z) \in T_eSO(3)$ is the angular velocity, v_O is the linear velocity, $\hat{\omega} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \in \mathbb{R}^{3\times 3}$, and the hat "$\wedge$" and unhat"$\vee$" are linear operators that relate the vector and matrix representations. The same representation and operators also apply to the spatial velocity $\overline{v} = (\overline{\omega}, \overline{v}_O) \in T_eSE(3)$, whose 6×1 vector and 4×4 matrix representations are respectively $\overline{v} = (\dot{g}g^{-1})^\vee$ and $\hat{\overline{v}} = \dot{g}g^{-1}$.

In the rest of this paper, if not specified, vector representation is used for $T_eSE(3)$, such as v, $\overline{v}$, etc., and the adjoint operators Ad_g and $\mathrm{ad}_v : T_eSE(3) \to T_eSE(3)$ can be accordingly represented as 6×6 matrices $\mathrm{Ad}_g = \begin{bmatrix} R & 0 \\ \hat{p}R & R \end{bmatrix}$ and $\mathrm{ad}_v = \begin{bmatrix} \hat{\omega} & 0 \\ \hat{v}_O & \hat{\omega} \end{bmatrix}$ such that $\overline{v} = \mathrm{Ad}_g v$ and $\mathrm{ad}_{v_1} v_2 = (\hat{v}_1\hat{v}_2 - \hat{v}_2\hat{v}_1)^\vee$. For consistence, the dual Lie algebra $T_e^*SE(3)$ uses the 6×1 vector representation as well. As a result, the body wrench $F = (\tau, f_O) \in T_e^*SE(3)$ is represented as a 6×1 vector $F = \begin{bmatrix}\tau^T & f_O^T\end{bmatrix}^T$ in which $\tau \in T_e^*SO(3)$ is the torque and f_O is the linear force so that $\langle F, v\rangle = F^T v$. Moreover, we define the linear operator $\mathrm{ad}_F^D : T_eSE(3) \to T_e^*SE(3)$ which is represented as a 6×6 matrix $\mathrm{ad}_F^D = \begin{bmatrix} \hat{\tau} & \hat{f}_O \\ \hat{f}_O & 0 \end{bmatrix}$ so that $F^T\mathrm{ad}_{v_1}v_2 = v_2^T\mathrm{ad}_F^D v_1 = -v_1^T\mathrm{ad}_F^D v_2$ for $v_1, v_2 \in T_eSE(3)$. The same representation and operators also apply to the spatial wrench $\overline{F} = \mathrm{Ad}_g^{-T}F = (\overline{\tau}, \overline{f}_O)$ which is paired with the spatial velocity $\overline{v} = \mathrm{Ad}_g v$.

2.3 The Tree Representation of Mechanical Systems

In general, a mechanical system with n inter-connected rigid bodies indexed as $1,\, 2,\, \cdots,\, n$ can be represented through a tree structure so that each rigid body has a single parent and zero or more children [2, 14], and such a representation is termed as *tree representation*. In this paper, the spatial frame is denoted as $\{0\}$, which is the root of the tree representation, and we denote the body frame of rigid body i as $\{i\}$, and the parent, ancestors, children and descendants of rigid body i as $\mathrm{par}(i)$, $\mathrm{anc}(i)$, $\mathrm{chd}(i)$ and $\mathrm{des}(i)$, respectively. Since all joints can be modeled using a combination of revolute joints and prismatic joints, we assume that each rigid body i is connected to its parent by a one-degree-of-freedom joint i which is either a revolute or a prismatic joint and parameterized by a real scalar $q_i \in \mathbb{R}$. As a result, the tree representation is parameterized with n generalized coordinates $q = \begin{bmatrix} q_1 \; q_2 \cdots q_n \end{bmatrix}^T \in \mathbb{R}^n$. For each joint i, the joint twist with respect to frame $\{0\}$ and $\{i\}$ are respectively denoted as 6×1 vectors $\overline{S}_i = \begin{bmatrix} \overline{s}_i^T \; \overline{n}_i^T \end{bmatrix}^T$ and $S_i = \begin{bmatrix} s_i^T \; n_i^T \end{bmatrix}^T$ in which $\overline{s}_i$, s_i are 3×1 vectors corresponding to rotation and $\overline{n}_i$, n_i are 3×1 vectors corresponding to translation. Note that S_i, s_i and n_i are constant by definition. Moreover, $\overline{S}_i$ and S_i are related as $\overline{S}_i = \mathrm{Ad}_{g_i} S_i$ where $g_i \in SE(3)$ is the configuration of rigid body i, and $\dot{\overline{S}}_i = \mathrm{ad}_{\overline{v}_i} \overline{S}_i$, where $\overline{v}_i \in T_e SE(3)$ is the spatial velocity of rigid body i.

It is assumed without loss of generality in this paper that the origin of frame $\{i\}$ is the mass center of rigid body i, and $j \in \mathrm{des}(i)$ only if $i < j$, or equivalently $j \in \mathrm{anc}(i)$ only if $i > j$.

The rigid body dynamics can be computed through the tree representation. The configuration $g_i = (R_i, p_i) \in SE(3)$ of rigid body i is $g_i = g_{\mathrm{par}(i)} g_{\mathrm{par}(i),i}(q_i)$ in which $g_{\mathrm{par}(i),i}(q_i) = g_{\mathrm{par}(i),i}(0) \exp(\hat{S}_i q_i)$ is the rigid body transformation from frame $\{i\}$ to its parent frame $\{\mathrm{par}(i)\}$, and the spatial velocity $\overline{v}_i$ of rigid body i is $\overline{v}_i = \overline{v}_{\mathrm{par}(i)} + \overline{S}_i \cdot \dot{q}_i$. In addition, the spatial inertia matrix $\overline{M}_i$ of rigid body $\{i\}$ with respect to frame $\{0\}$ is $\overline{M}_i = \mathrm{Ad}_{g_i}^{-T} M_i \mathrm{Ad}_{g_i}^{-1}$ in which $M_i = \mathrm{diag}\{\mathcal{I}_i, m_i \mathbf{I}\} \in \mathbb{R}^{6\times 6}$ is the constant body inertia matrix of rigid body i, $\mathcal{I}_i \in \mathbb{R}^{3\times 3}$ is the body rotational inertia matrix, $m_i \in \mathbb{R}$ is the mass and $\mathbf{I} \in \mathbb{R}^{3\times 3}$ is the identity matrix.

In rigid body dynamics, an important notion is the *articulated body* [14]. In terms of the tree representation, articulated rigid body i consists of rigid body i and all its descendants $j \in \mathrm{des}(i)$, and the interactions with articulated body i can only be made through rigid body i, which is known as the handle of the articulated body i.

In the last thirty years, a number of algorithms for efficiently computing the rigid body dynamics have been developed based on tree representations and articulated bodies [14–16], making explicit integrators have $O(n)$ complexity for an n-degree-of-freed-om mechanical system. Even though the same algorithms might be used for the evaluation of implicit integrators, *none of them can be used for the computation of the Newton direction for solving implicit integrators*. If the residue is r^k, the Newton direction of an implicit integrator is computed as $\delta q^k = -\mathcal{J}(q^k)^{-1} r^k$; however, the Jacobian matrix $\mathcal{J}(q^k)$ is usually asymmetric and indefinite, and has a size greater than $n \times n$ for higher-order implicit integrators, which means that the computation of implicit integrators is distinct from explicit integrators whose computation is simply a combination of the algorithms in [14–16] with an appropriate integration scheme. Furthermore, the computation of implicit integrators is much more complicated than the computation of forward and inverse dynamics and out of the scope of those algorithms in [14–16].

3 The Linear-Time Higher-Order Variational Integrator

In this and next section, we present the propositions and algorithms efficiently computing higher-order variational integrators, whose derivations are omitted due to space limitations. Though not required for implementation, we refer the reader to the supplementary appendix of this paper [17] for detailed proofs.[1]

In the rest of this paper, if not specified, we assume that the mechanical system has n degrees of freedom and the higher-order variational integrator has $s+1$ control points $q^{k,\alpha} = q(t^{k,\alpha})$ in which $0 \le \alpha \le s$. Note that the notation $(\cdot)^{k,\alpha}$ is used to denote quantities $(\cdot)$ associated with $q^{k,\alpha}$ and $t^{k,\alpha}$, such as $q_i^{k,\alpha}$, $g_i^{k,\alpha}$, $\overline{v}_i^{k,\alpha}$, etc.

3.1 The DEL Equation Evaluation

To evaluate the DEL equations, the *discrete articulated body momentum* and *discrete articulated body impulse* are defined from the perspective of articulated bodies as follows.

Definition 1. *The discrete articulated body momentum $\overline{\mu}_i^{k,\alpha} \in \mathbb{R}^6$ for articulated body i is defined to be $\overline{\mu}_i^{k,\alpha} = \overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)} \overline{\mu}_j^{k,\alpha}$ in which $\overline{M}_i^{k,\alpha}$ and $\overline{v}_i^{k,\alpha}$ are respectively the spatial inertia matrix and spatial velocity of rigid body i.*

Definition 2. *Suppose $\overline{F}_i(t) \in \mathbb{R}^6$ is the sum of all the wrenches directly acting on rigid body i, which does not include those applied or transmitted through the joints that are connected to rigid body i. The discrete articulated body impulse $\overline{\Gamma}_i^{k,\alpha} \in \mathbb{R}^6$ for articulated body i is defined to be $\overline{\Gamma}_i^{k,\alpha} = \overline{F}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)} \overline{\Gamma}_j^{k,\alpha}$ in which $\overline{F}_i^{k,\alpha} = \omega^\alpha \overline{F}_i(t^{k,\alpha})\Delta t \in \mathbb{R}^6$ is the discrete impulse acting on rigid body i. Note that $\overline{F}_i(t)$, $\overline{F}_i^{k,\alpha}$ and $\overline{\Gamma}_i^{k,\alpha}$ are expressed in frame $\{0\}$.*

Remark 1. As for wrenches exerted on rigid body i, in addition to $\overline{F}_i(t)$ which includes gravity as well as the external wrenches that directly act on rigid body i, there are also wrenches applied through joints, e.g., from actuators, and wrenches transmitted through joints, e.g., from the parent and children of rigid body i in the tree representation.

It can be seen in Proposition 1 that $\overline{\mu}_i^{k,\alpha}$ and $\overline{\Gamma}_i^{k,\alpha}$ make it possible to evaluate the DEL equations without explicitly calculating $\mathbb{D}_{\alpha+1}\mathcal{L}_d(\overline{q}^k)$ and $\mathcal{F}_d^{k,\alpha}$ in Eqs. (5) and (6).

Proposition 1. *If $Q_i(t) \in \mathbb{R}$ is the sum of all joint forces applied to joint i and $p^k = \begin{bmatrix} p_1^k \ p_2^k \ \cdots \ p_n^k \end{bmatrix}^T \in \mathbb{R}^n$ is the discrete momentum, the DEL equations Eq. (5) can be evaluated as*

$$r_i^{k,0} = p_i^k + \overline{S}_i^{k,0^T} \cdot \overline{\Omega}_i^{k,0} + \sum_{\beta=0}^{s} a^{0\beta}\overline{S}_i^{k,\beta^T} \cdot \overline{\mu}_i^{k,\beta} + Q_i^{k,0}, \tag{7a}$$

$$r_i^{k,\alpha} = \overline{S}_i^{k,\alpha^T} \cdot \overline{\Omega}_i^{k,\alpha} + \sum_{\beta=0}^{s} a^{\alpha\beta}\overline{S}_i^{k,\beta^T} \cdot \overline{\mu}_i^{k,\beta} + Q_i^{k,\alpha} \quad \forall\alpha = 1, \cdots, s-1, \tag{7b}$$

$$p_i^{k+1} = \overline{S}_i^{k,s^T} \cdot \overline{\Omega}_i^{k,s} + \sum_{\beta=0}^{s} a^{s\beta}\overline{S}_i^{k,\beta^T} \cdot \overline{\mu}_i^{k,\beta} + Q_i^{k,s} \tag{7c}$$

[1] In addition to the proofs, the supplementary appendix [17] also contains the complete $O(n)$ algorithms to compute the Newton direction for higher-order variational integrators.

Algorithm 1 Recursive Evaluation of the DEL Equations

initialize $g_0^{k,\alpha} = \mathbf{I}$ and $\overline{v}_0^{k,\alpha} = 0$
for $i = 1 \to n$ **do**
    for $\alpha = 0 \to s$ **do**
        $g_i^{k,\alpha} = g_{\mathrm{par}(i)}^{k,\alpha} g_{\mathrm{par}(i),i}^{k,\alpha}(q_i^{k,\alpha})$
        $\overline{S}_i^{k,\alpha} = \mathrm{Ad}_{g_i^{k,\alpha}} S_i, \quad \overline{M}_i^{k,\alpha} = \mathrm{Ad}_{g_i^{k,\alpha}}^{-T} M_i \mathrm{Ad}_{g_i^{k,\alpha}}^{-1}$
        $\dot{q}_i^{k,\alpha} = \frac{1}{\Delta t} \sum\limits_{\beta=0}^{s} b^{\alpha\beta} q_i^{k,\beta}, \quad \overline{v}_i^{k,\alpha} = \overline{v}_{\mathrm{par}(i)}^{k,\alpha} + \overline{S}_i^{k,\alpha} \cdot \dot{q}_i^{k,\alpha}$
    end for
end for
for $i = n \to 1$ **do**
    for $\alpha = 0 \to s$ **do**
        $\overline{\mu}_i^{k,\alpha} = \overline{M}_i^{k,\alpha} \overline{v}_i^{k,\alpha} + \sum\limits_{j \in \mathrm{chd}(i)} \overline{\mu}_j^{k,\alpha}, \quad \overline{\mathit{\Gamma}}_i^{k,\alpha} = \overline{F}_i^{k,\alpha} + \sum\limits_{j \in \mathrm{chd}(i)} \overline{\mathit{\Gamma}}_j^{k,\alpha}$
        $\overline{\mathit{\Omega}}_i^{k,\alpha} = w^\alpha \mathit{\Delta} t \cdot \mathrm{ad}_{\overline{v}_i^{k,\alpha}}^T \cdot \overline{\mu}_i^{k,\alpha} + \overline{\mathit{\Gamma}}_i^{k,\alpha}$
    end for
    $r_i^{k,0} = p_i^k + \overline{S}_i^{k,0\,T} \overline{\mathit{\Omega}}_i^{k,0} + \sum\limits_{\beta=0}^{s} a^{0\beta} \overline{S}_i^{k,\beta\,T} \cdot \overline{\mu}_i^{k,\beta} + Q_i^{k,0}$
    for $\alpha = 1 \to s-1$ **do**
        $r_i^{k,\alpha} = \overline{S}_i^{k,\alpha\,T} \overline{\mathit{\Omega}}_i^{k,\alpha} + \sum\limits_{\beta=0}^{s} a^{\alpha\beta} \overline{S}_i^{k,\beta\,T} \cdot \overline{\mu}_i^{k,\beta} + Q_i^{k,\alpha}$
    end for
    $p_i^{k+1} = \overline{S}_i^{k,s\,T} \overline{\mathit{\Omega}}_i^{k,s} + \sum\limits_{\beta=0}^{s} a^{s\beta} \overline{S}_i^{k,\beta\,T} \cdot \overline{\mu}_i^{k,\beta} + Q_i^{k,s}$
end for

in which $r_i^{k,\alpha}$ *is the residue of the DEL equations Eqs.* (5a) *and* (5b), $a^{\alpha\beta} = w^\beta b^{\beta\alpha}$, $\overline{\mathit{\Omega}}_i^{k,\alpha} = w^\alpha \mathit{\Delta} t \cdot \mathrm{ad}_{\overline{v}_i^{k,\alpha}}^T \cdot \overline{\mu}_i^{k,\alpha} + \overline{\mathit{\Gamma}}_i^{k,\alpha}$, *and* $Q_i^{k,\alpha} = \omega^\alpha Q_i(t^{k,\alpha}) \mathit{\Delta} t$ *is the discrete joint force applied to joint* i.

Proof. See [17, Section D.1] □

In Eqs. (7a) and (7b), if all $r_i^{k,\alpha}$ are equal to zero, a solution to the variational integrator as well as the DEL equations is obtained.

All the quantities used in Proposition 1 can be recursively computed in the tree representation, therefore, we have Algorithm 1 that evaluates the DEL equations, which essentially consists of $s+1$ forward passes from root to leaf nodes and $s+1$ backward passes in the reverse order, thus totally takes $O(sn)$ time. In contrast, automatic differentiation and our prior methods [2,4] take $O(sn^2)$ time to evaluate the DEL equations.

3.2 Exact Newton Direction Computation

From Eq. (5), the Newton direction $\delta\overline{q}^k = \left[\delta q^{k,1\,T}, \cdots, \delta q^{k,s\,T}\right]^T \in \mathbb{R}^{sn}$ is computed as $\delta\overline{q}^k = -\mathcal{J}^{k^{-1}}(\overline{q}^k) \cdot r^k$ in which $\mathcal{J}^k(\overline{q}^k) \in \mathbb{R}^{sn \times sn}$ is the Jacobian of Eqs. (5a) and (5b) with respect to control points $q^{k,1}, \cdots, q^{k,s}$, and $r^k \in \mathbb{R}^{sn}$ is the residue of evaluating the DEL equations Eqs. (5a) and (5b) by Proposition 1.

In this section, we make the the following assumption on $\overline{F}_i^{k,\alpha}$ and $Q_i^{k,\alpha}$, which is general and applies to a large number of mechanical systems in robotics.

Assumption 1. *Let $u(t)$ be the control inputs of the mechanical system, we assume that the discrete impulse $\overline{F}_i^{k,\alpha}$ and discrete joint force $Q_i^{k,\alpha}$ can be respectively formulated as $\overline{F}_i^{k,\alpha} = \overline{F}_i^{k,\alpha}(g_i^{k,\alpha}, \overline{v}_i^{k,\alpha}, u^{k,\alpha})$ and $Q_i^{k,\alpha} = Q_i^{k,\alpha}(q_i^{k,\alpha}, \dot{q}_i^{k,\alpha}, u^{k,\alpha})$ in which $u^{k,\alpha} = u(t^{k,\alpha})$.*

If Assumption 1 holds and ${\mathcal{J}^k}^{-1}(\overline{q}^k)$ exists, it can be shown that [17, Algorithm B.1] computes the Newton direction for variational integrators in $O(s^3 n)$ time.

Proposition 2. *For higher-order variational integrators of unconstrained mechanical systems, if Assumption 1 holds and ${\mathcal{J}^k}^{-1}(\overline{q}^k)$ exists, the Newton direction $\delta\overline{q}^k = -{\mathcal{J}^k}^{-1}(\overline{q}^k) \cdot r^k$ can be computed with [17, Algorithm B.1] in $O(s^3 n)$ time.*

Proof. See [17, Section D.2]. □

In [17, Algorithm B.1], the forward and backward passes of the tree structure take $O(s^2 n)$ time, and the n computations of the $s \times s$ matrix inverse takes $O(s^3 n)$ time, thus the overall complexity of [17, Algorithm B.1] is $O(s^3 n + s^2 n)$. In contrast, automatic differentiation and our prior methods in [2, 4] take $O(s^2 n^3)$ time to compute $\mathcal{J}^k(\overline{q}^k)$ and another $O(s^3 n^3)$ time to compute the $sn \times sn$ matrix inverse ${\mathcal{J}^k}^{-1}(\overline{q}^k)$, and the overall complexity is $O(s^3 n^3 + s^2 n^3)$. Though the quasi-Newton method [9] is $O(n)$ time for second-order variational integrator in which $s = 1$, it requires small time steps and can not be used for third- or higher-order variational integrators.

Therefore, both Algorithm 1 and [17, Algorithm B.1] have $O(n)$ complexity for a given s, which results in a linear-time variational integrator. Furthermore, Algorithm 1 and [17, Algorithm B.1] have no restrictions on the number of control points, which indicates that the resulting linear-time variational integrator can be arbitrarily high order. *To our knowledge, this is the first exactly linear-time third- or higher-order implicit integrator for mechanical systems.*

3.3 Extension to Constrained Mechanical Systems

Thus far all our discussions of linear-time variational integrators have been restricted to unconstrained mechanical systems. However, Algorithm 1 and [17, Algorithm B.1] can be extended to constrained mechanical systems as well.

In terms of the the DEL equation evaluation, the extension to constrained mechanical systems is immediate. From Eq. (6), we only need to add the constraint term $A^{k,\alpha}(q^{k,\alpha}) \cdot \lambda^{k,\alpha}$ to the results of using Algorithm 1.

If the variational integrator is second-order and the mechanical system has m constraints, it is possible to compute the Newton direction δq^{k+1} and $\delta\lambda^k$ in $O(mn) + O(m^3)$ time using [17, Algorithm B.1]. In accordance with Eq. (6), δq^{k+1} and $\delta\lambda^k$ should satisfy $\mathcal{J}^k(q^k) \cdot \delta q^{k+1} + A^k(q^k) \cdot \delta\lambda^k = -r_q^k$ and $\mathbb{D}h^k(q^{k+1}, \dot{q}^{k+1}) \cdot \delta q^{k+1} =$

$-r_c^k$ in which r_q^k and r_c^k are equation residues. Then δq^{k+1} and $\delta\lambda^k$ can be computed as follows: i) compute $\delta q_r^{k+1} = -\mathcal{J}^{k^{-1}} \cdot r_q^k$ with [17, Algorithm B.1] which takes $O(n)$ time; ii) compute $\mathcal{J}^{k^{-1}} \cdot A^k$ by using [17, Algorithm B.1] m times which takes $O(mn)$ time; iii) compute $\delta\lambda^k = \left(\mathbb{D}h^k \cdot \mathcal{J}^k \cdot A^k\right)^{-1}(r_c^k + \mathbb{D}h^k \cdot \delta q_r^{k+1})$ which takes $O(m^3)$ time; iv) compute $\delta q^{k+1} = \delta q_r^{k+1} - \mathcal{J}^{k^{-1}} \cdot A^k \cdot \delta\lambda^k$.

In regard to third- or higher-order variational integrators, if the constraints are of $h_i^k(g_i^{k,\alpha}, \overline{v}_i^{k,\alpha}) = 0$ or $h_i^k(q_i^{k,\alpha}, \dot{q}_i^{k,\alpha}) = 0$ or both for each $i = 1, 2, \cdots, n$, [17, Algorithm B.1] can be used to compute the Newton direction $\delta\overline{q}^k$ and $\delta\lambda^k$ in a similar procedure to the second-order variational integrator.

In next section, we will discuss the linearization of higher-order variational integrators in $O(n^2)$ time.

4 The Linearization of Higher-Order Variational Integrators

The linearization of discrete time systems is useful for trajectory optimization, stability analysis, controller design, etc., which are import tools in robotics.

From Eqs. (5) and (6), the linearization of variational integrators is comprised of the computation of $\mathbb{D}^2\mathcal{L}_d(\overline{q}^k)$, $\mathbb{D}\mathcal{F}_d^{k,\alpha}(t^{k,\alpha})$ and $\mathbb{D}A^{k,\alpha}(q^{k,\alpha})$. In most cases, $\mathbb{D}\mathcal{F}_d^{k,\alpha}(t^{k,\alpha})$ and $\mathbb{D}A^{k,\alpha}(q^{k,\alpha})$ can be efficiently computed in $O(n^2)$ time, therefore, the linearization efficiency is mostly affected by $\mathbb{D}^2\mathcal{L}_d(\overline{q}^k)$.

It is by definition that the Lagrangian of a mechanical system is $\mathcal{L}(q,\dot{q}) = K(q,\dot{q}) - V(q)$ in which $K(q,\dot{q})$ is the kinetic energy and $V(q)$ is the potential energy, and from Eq. (4), the computation of $\mathbb{D}^2\mathcal{L}_d(\overline{q}^k)$ is actually to compute $\frac{\partial K}{\partial \dot{q}^2}$, $\frac{\partial^2 K}{\partial \dot{q}\partial q}$, $\frac{\partial^2 K}{\partial q\partial \dot{q}}$, $\frac{\partial^2 K}{\partial q^2}$ and $\frac{\partial V}{\partial q^2}$, for which we have Proposition 3 and Proposition 4 as follows.

Proposition 3. *For the kinetic energy $K(q,\dot{q})$ of a mechanical system, $\frac{\partial^2 K}{\partial \dot{q}^2}$, $\frac{\partial^2 K}{\partial \dot{q}\partial q}$, $\frac{\partial^2 K}{\partial q\partial \dot{q}}$, $\frac{\partial^2 K}{\partial q^2}$ can be recursively computed with Algorithm 2 in $O(n^2)$ time.*

Proof. See [17, Section D.3]. □

In the matter of potential energy $V(q)$, we only consider the gravitational potential energy $V_{\mathbf{g}}(q)$, and the other types of potential energy can be computed in a similar way.

Proposition 4. *If $\mathbf{g} \in \mathbb{R}^3$ is gravity, then for the gravitational potential energy $V_{\mathbf{g}}(q)$, $\frac{\partial^2 V_{\mathbf{g}}}{\partial q^2}$ can be recursively computed with Algorithm 3 in $O(n^2)$ time.*

Proof. See [17, Section D.4]. □

In regard to Proposition 4 and Algorithm 3, we remind the reader of the notation introduced in Sections 2.2 and 2.3 that $m_i \in \mathbb{R}$ is the mass of rigid body i, $p_i \in \mathbb{R}^3$ is the mass center of rigid body i as well as the origin of frame $\{i\}$, and $\overline{S}_i = \left[\overline{s}_i^T \ \overline{n}_i^T\right]^T \in \mathbb{R}^6$ is the spatial Jacobian of joint i with respect to frame $\{0\}$.

If $\frac{\partial K}{\partial \dot{q}^2}$, $\frac{\partial^2 K}{\partial \dot{q}\partial q}$, $\frac{\partial^2 K}{\partial q\partial \dot{q}}$, $\frac{\partial^2 K}{\partial q^2}$ and $\frac{\partial V}{\partial q^2}$ are computed in $O(n^2)$ time, then according to Eqs. (2) and (4), the remaining computation of $\mathbb{D}^2\mathcal{L}_d(\overline{q}^{k,\alpha})$ is simply the application

Algorithm 2 Recursive Computation of $\frac{\partial^2 K}{\partial \dot{q}^2}, \frac{\partial^2 K}{\partial \dot{q}\partial q}, \frac{\partial^2 K}{\partial q \partial \dot{q}}, \frac{\partial^2 K}{\partial q^2}$

initialize $g_0 = \mathbf{I}$ and $\overline{v}_0 = 0$
for $i = 1 \to n$ **do**
  $g_i = g_{\mathrm{par}(i)} g_{\mathrm{par}(i),i}(q_i)$
  $\overline{M}_i = \mathrm{Ad}_{g_i}^{-T} M_i \mathrm{Ad}_{g_i}^{-1}, \quad \overline{S}_i = \mathrm{Ad}_{g_i} S_i$
  $\overline{v}_i = \overline{v}_{\mathrm{par}(i)} + \overline{S}_i \cdot \dot{q}_i, \quad \dot{\overline{S}}_i = \mathrm{ad}_{\overline{v}_i} \overline{S}_i$
end for
initialize $\frac{\partial^2 K}{\partial \dot{q}^2} = \mathbf{0}, \quad \frac{\partial^2 K}{\partial \dot{q}\partial q} = \mathbf{0}, \quad \frac{\partial^2 K}{\partial q \partial \dot{q}} = \mathbf{0}, \quad \frac{\partial^2 K}{\partial q^2} = \mathbf{0}$
for $i = n \to 1$ **do**
  $\overline{\mu}_i = \overline{M}_i \overline{v}_i + \sum_{j \in \mathrm{chd}(i)} \overline{\mu}_j, \quad \overline{\mathcal{M}}_i = \overline{M}_i + \sum_{j \in \mathrm{chd}(i)} \overline{\mathcal{M}}_j$
  $\overline{\mathcal{M}}_i^A = \overline{\mathcal{M}}_i \overline{S}_i, \quad \overline{\mathcal{M}}_i^B = \overline{\mathcal{M}}_i \dot{\overline{S}}_i - \mathrm{ad}_{\overline{\mu}_i}^D \overline{S}_i$
  for $j \in \mathrm{anc}(i) \cup \{i\}$ **do**
    $\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j} = \frac{\partial^2 K}{\partial \dot{q}_j \partial \dot{q}_i} = \overline{S}_j^T \overline{\mathcal{M}}_i^A$
    $\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j} = \frac{\partial^2 K}{\partial q_j \partial \dot{q}_i} = \dot{\overline{S}}_j^T \overline{\mathcal{M}}_i^A, \quad \frac{\partial^2 K}{\partial q_i \partial \dot{q}_j} = \frac{\partial^2 K}{\partial \dot{q}_j \partial q_i} = \overline{S}_j^T \overline{\mathcal{M}}_i^B$
    $\frac{\partial^2 K}{\partial q_i \partial q_j} = \frac{\partial^2 K}{\partial q_j \partial q_i} = \dot{\overline{S}}_j^T \overline{\mathcal{M}}_i^B$
  end for
end for

Algorithm 3 Recursive Computation of $\frac{\partial^2 V_{\mathbf{g}}}{\partial q^2}$

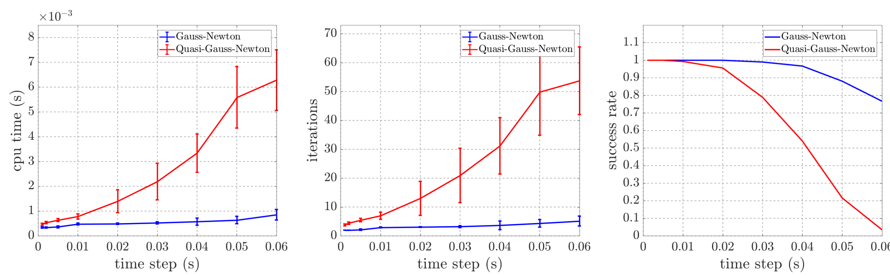
initialize $g_0 = \mathbf{I}$
for $i = 1 \to n$ **do**
  $g_i = g_{\mathrm{par}(i)} g_{\mathrm{par}(i),i}(q_i), \quad \overline{S}_i = \mathrm{Ad}_{g_i} S_i$
end for
initialize $\frac{\partial^2 V_{\mathbf{g}}}{\partial q^2} = \mathbf{0}$
for $i = n \to 1$ **do**
  $\overline{\sigma}_{m_i} = m_i + \sum_{j \in \mathrm{chd}(i)} \overline{\sigma}_{m_j}, \quad \overline{\sigma}_{p_i} = m_i p_i + \sum_{j \in \mathrm{chd}(i)} \overline{\sigma}_{p_j}$
  $\overline{\sigma}_i^A = \hat{\mathbf{g}} \left(\overline{\sigma}_{m_i} \cdot \overline{n}_i - \hat{\overline{\sigma}}_{p_i} \cdot \overline{s}_i \right)$
  for $j \in \mathrm{anc}(i) \cup \{i\}$ **do**
    $\frac{\partial^2 V_{\mathbf{g}}}{\partial q_i \partial q_j} = \frac{\partial^2 V_{\mathbf{g}}}{\partial q_j \partial q_i} = \overline{s}_j^T \cdot \overline{\sigma}_i^A$
  end for
end for

of the chain rule. Therefore, if the variational integrator has $s + 1$ control points, the complexity of the linearization is $O(s^2 n^2)$. In contrast, automatic differentiation and our prior methods [2, 4] take $O(s^2 n^3)$ time to linearize the variational integrators.

5 Comparison with Existing Methods

The variational integrators using Algorithms 1 to 3 and [17, Algorithm B.1] are compared with the linear-time quasi-Newton method [9], automatic differentiation and the Hermite-Simpson direct collocation method, which verifies the accuracy, efficiency and scalability of our work. All the tests are run in C++ on a 3.1GHz Intel Core Xeon Thinkpad P51 laptop.

5.1 Comparison with the Linear-Time Quasi-Newton Method



(a) (b) (c)

Fig. 1: The comparison of the $O(n)$ Newton method with the $O(n)$ quasi-Newton method [9] for the trapezoidal variational integrator of a 32-link pendulum with different time steps. The results of computational time are in (a), number of iterations in (b) and success rates in (c). Each result is calculated over 1000 initial conditions.

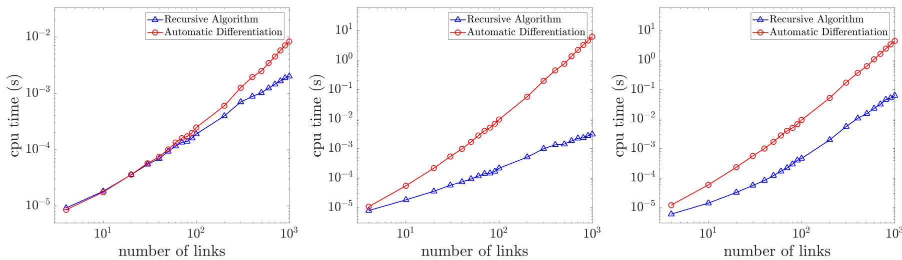
In this subsection, we compare the $O(n)$ Newton method using Algorithm 1 and [17, Algorithm B.1] with the $O(n)$ quasi-Newton method in [9] on the trapezoidal variational integrator (Example 1) of a 32-link pendulum with different time steps.

In the comparison, 1000 initial joint angles q^0 and joint velocities $\dot{q}^0$ are uniformly sampled from $[-\frac{\pi}{2}, \frac{\pi}{2}]$ for each of the selected time steps, which are 0.001s, 0.002s, 0.005s, 0.01s, 0.02s, 0.03s, 0.04s, 0.05s and 0.06s, and the Newton and quasi-Newton methods are used to solve the DEL equations for one time step. The results are in Fig. 1, in which the computational time and the number of iterations are calculated only over initial conditions that the DEL equations are successfully solved. It can be seen that the Newton method using Algorithm 1 and [17, Algorithm B.1] outperforms the quasi-Newton method in [9] in all aspects, especially for relatively large time steps.

5.2 Comparison with Automatic Differentiation

In this subsection, we compare Algorithms 1 to 3 and [17, Algorithm B.1] with automatic differentiation for evaluating the DEL equations, computing the Newton direction and linearizing the DEL equations. The variational integrator used is the Simpson variational integrator (Example 2).

In the comparison, we use pendulums with different numbers of links as benchmark systems. For each pendulum, 100 initial joint angles q^0 and joint velocities $\dot{q}^0$ are uniformly sampled from $[-\frac{\pi}{2}, \frac{\pi}{2}]$. The results are in Fig. 2 and it can be seen that our recursive algorithms are much more efficient, which is consistent with the fact that Algorithms 1 to 3 and [17, Algorithm B.1] are $O(n)$ for evaluating the DEL equations, $O(n)$ for computing the Newton direction, and $O(n^2)$ for linearizing the DEL equations, whereas automatic differentiation are $O(n^2)$, $O(n^3)$ and $O(n^3)$, respectively.



(a)

(b)

(c)

Fig. 2: The comparison of our recursive algorithms with automatic differentiation for pendulums with different numbers of links. The variational integrator used is the Simpson variational integrator. The results of evaluating the DEL equations are in (a), computing the Newton direction in (b) and linearizing the DEL equations in (c). Each result is calculated over 100 initial conditions.

5.3 Comparison with the Hermite-Simpson Direct Collocation Method

In this subsection, we compare the fourth-order Simpson variational integrator (Example 2) with the Hermite-Simpson direct collocation method, which is a third-order implicit integrator commonly used in robotics for trajectory optimization [10,11].[2] Note that both integrators use three control points for integration.

The strict comparison of the two integrators for trajectory optimization is usually difficult since it depends on a number of factors, such as the target problem, the optimizers used, the optimality and feasibility tolerances, etc. Therefore, we compare the Simpson variational integrator and the Hermite-Simpson direct collocation method by listing the order of accuracy, the number of variables and the number of constraints for trajectory optimization. In general, the computational loads of optimization depends on the problem size that is directly related with the number of variables and the the number of constraints. The higher-order accuracy suggests the possibility of large time steps in trajectory optimization, which reduces not only the problem size but the computational loads of optimization as well. The results are in Table 1.[3] It can be concluded that the Simpson variational integrator is more accurate and has less variables and constraints in trajectory optimization, especially for constrained mechanical systems.

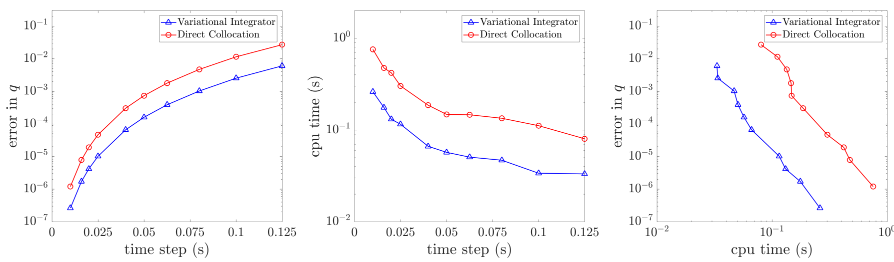
The accuracy comparison in Table 1 of the Simpson variational integrator with the Hermite-Simpson direct collocation method is further numerically validated on a 12-link pendulum. In the comparison, different time steps are used to simulate 100 trajectories with the final time $T = 10$ s, and the initial joint angles q^0 are uniformly sam-

[2] The Hermite-Simpson direct collocation methods used in [10, 11] are actually implicit integrators that integrate the trajectory as a second-order system in the $(q, \dot{q})$ space, whereas the variational integrators integrate the trajectory in the (q, p) space.

[3] The explicit and implicit formulations of the Hermite-Simpson direct collocation methods differ in whether the joint acceleration $\ddot{q}$ is explicitly computed or implicitly involved as extra variables. Even though the explicit formulation of the Hermite-Simpson direct collocation has less variables and constraints than the implicit formulation, it is usually more complicated for the evaluation and linearization, therefore, the implicit formulation is usually more efficient and more commonly used in trajectory optimization [11].

integrator	accuracy	# of variables	# of constraints
variational integrator	4th-order	$(4N+3)n+(2N+1)m$	$3Nn+(2N+1)m$
direct collocation (explicit)	3rd-order	$(6N+3)n+(2N+1)m$	$4Nn+(6N+3)m$
direct collocation (implicit)	3rd-order	$(8N+4)n+(2N+1)m$	$(6N+1)n+(6N+3)m$

Table 1: The comparison of the Simpson variational integrator with the Hermite-Simpson direct collocation method for trajectory optimization. The trajectory optimization problem has N stages and the mechanical system has n degrees of freedom, m holonomic constraints and is fully actuated with n control inputs. Note that both integrators use three control points for integration.



(a) (b) (c)

Fig. 3: The comparison of the Simpson variational integrator with the Hermite-Simpson direction collocation method on a 12-link pendulum with different time steps. The results of the integrator error are in (a), the computational time in (b) and the integration error v.s. computational time in (c). Each result is calculated over 100 initial conditions.

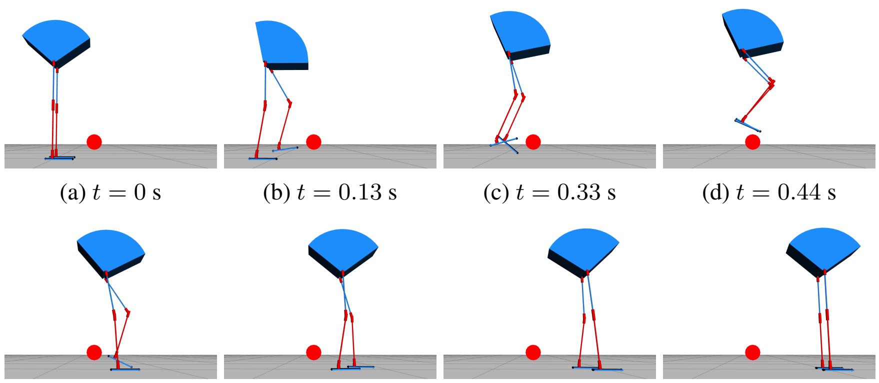
pled from $[-\frac{\pi}{12}, \frac{\pi}{12}]$ and the initial joint velocities $\dot{q}^0$ are zero. Moreover, the Simpson variational integrator uses Algorithm 1 and [17, Algorithm B.1] which has $O(n)$ complexity for the integrator evaluation and the Newton direction computation, whereas the Hermite-Simpson direct collocation method uses [14, 18] which is $O(n)$ for the integrator evaluation and $O(n^3)$ for the Newton direction computation. For each initial condition, the benchmark solution $q_d(t)$ is created from the Hermite-Simpson direct collocation method with a time step of 5×10^{-4} s and the simulation error in $q(t)$ is evaluated as $\frac{1}{T}\int_0^T \|q(t)-q_d(t)\|dt$. The running time of the simulation is also recorded. The results are in Fig. 3, which indicates that the Simpson variational integrator is more accurate and more efficient in simulation, and more importantly, a better alternative to the Hermite-Simpson direction collocation method for trajectory optimization.

In regard to the integrator evaluation and linearization, for unconstrained mechanical systems, experiments (not shown) suggest that the Simpson variational integrator using Algorithms 1 to 3 is usually faster than the Hermite-Simpson direct collocation method using [14, 18] even though theoretically both integrators have the same order of complexity. However, for constrained mechanical systems, if there are m holonomic constraints, the Simpson variational integrator is $O(mn)$ for the evaluation and $O(mn^2)$ for the linearization while the Hermite-Simpson direct collocation method in [10, 11] is respectively $O(mn^2)$ and $O(mn^3)$, the difference of which results from that the Hermite-Simpson direct collocation method is more complicated to model the constrained dynamics.

6 Implementation for Trajectory Optimization

In this section, we implement the fourth-order Simpson variational integrator (Example 2) with Algorithms 1 to 3 on the Spring Flamingo robot [19], the LittleDog robot [20] and the Atlas robot [21] for trajectory optimization, the results of which are included in our supplementary videos. It should be noted that the variational integrators used in [2–4, 6, 8] for trajectory optimization are second order. In Sections 6.1 and 6.2, a LCP formulation similar to [8] is used to model the discontinuous frictional contacts with which no contact mode needs to be prespecified. These examples indicate that higher-order variational integrators are good alternatives to the direct collocation methods [10, 11]. The trajectory optimization problems are solved with SNOPT [22].

6.1 Spring Flamingo



(a) $t = 0$ s (b) $t = 0.13$ s (c) $t = 0.33$ s (d) $t = 0.44$ s

(e) $t = 0.57$ s (f) $t = 0.68$ s (g) $t = 0.88$ s (h) $t = 1.1$ s

Fig. 4: The Spring Flamingo robot jumps over a obstacle of 0.16 meters high.

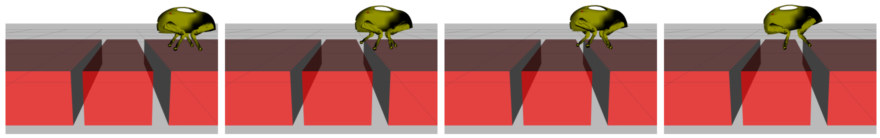
The Spring Flamingo robot is a 9-DoF flat-footed biped robot with actuated hips and knees and passive springs at ankles [19]. In this example, the Spring Flamingo robot is commanded to jump over an obstacle that is 0.16 m high while walking horizontally from one position to another. The results are in Fig. 4, in which the initial walking velocity is 0.26 m/s and the average walking velocity is around 0.9 m/s.

6.2 LittleDog

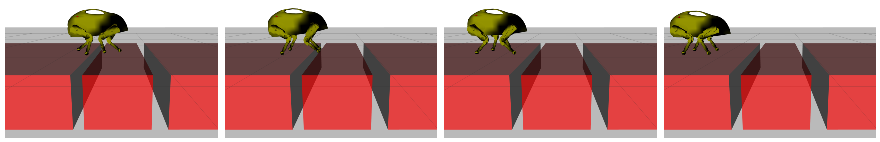
The LittleDog robot is 18-DoF quadruped robot used in research of robot walking [20]. In this example, the LittleDog robot is required to walk over terrain with two gaps. The results are in Fig. 5, in which the average walking velocity is 0.25 m/s.

6.3 Atlas

The Atlas robot is a 30-DoF humanoid robot used in the DARPA Robotics Challenge [21]. In this example, the Atlas robot is required to pick a red ball with its left

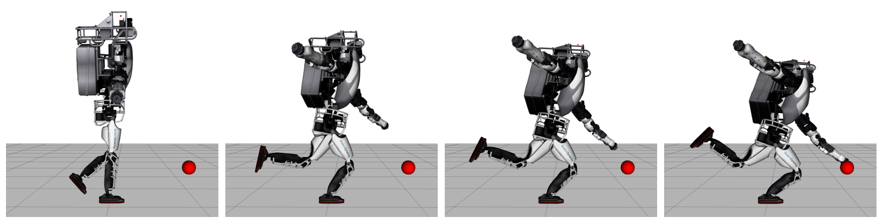


(a) $t = 0$ s (b) $t = 0.48$ s (c) $t = 0.56$ s (d) $t = 1.04$ s



(e) $t = 1.84$ s (f) $t = 2.16$ s (g) $t = 2.72$ s (h) $t = 3.2$ s

Fig. 5: The LittleDog robot walks over terrain with gaps.



(a) $t = 0$s (b) $t = 0.4$ s (c) $t = 0.6$ s (d) $t = 1.3$ s

Fig. 6: The Atlas robot picks a red ball while keeping balanced with a single foot.

hand while keeping balanced only with its right foot. Moreover, the contact wrenches applied to the supporting foot should satisfy contact constraints of a flat foot [11]. The results are in Fig. 6 and it takes around 1.3 s for the Atlas robot to pick the ball.

7 Conclusion

In this paper, we present $O(n)$ algorithms for the linear-time higher-order variational integrators and $O(n^2)$ algorithms to linearize the DEL equations for use in trajectory optimization. The proposed algorithms are validated through comparison with existing methods and implementation on robotic systems for trajectory optimization. The results illustrate that the same integrator can be used for simulation and trajectory optimization in robotics, preserving mechanical properties while achieving good scalability and accuracy. Furthermore, thought not presented in this paper, these $O(n)$ algorithms can be regularized for parallel computation, which results in $O(\log(n))$ algorithms with enough processors.

References

1. Jerrold E Marsden and Matthew West. Discrete mechanics and variational integrators. *Acta Numerica*, 10:357–514, 2001.

2. Elliot R Johnson and Todd D Murphey. Scalable variational integrators for constrained mechanical systems in generalized coordinates. *IEEE Transactions on Robotics*, 25(6):1249–1261, 2009.
3. Marin Kobilarov, Keenan Crane, and Mathieu Desbrun. Lie group integrators for animation and control of vehicles. *ACM Transactions on Graphics (TOG)*, 28(2):16, 2009.
4. Elliot Johnson, Jarvis Schultz, and Todd Murphey. Structured linearization of discrete mechanical systems for analysis and optimal control. *IEEE Transactions on Automation Science and Engineering*, 2015.
5. Taosha Fan and Todd Murphey. Structured linearization of discrete mechanical systems on lie groups: A synthesis of analysis and control. In *IEEE Conference on Decision and Control (CDC)*, pages 1092–1099, 2015.
6. Oliver Junge, Jerrold E Marsden, and Sina Ober-Blöbaum. Discrete mechanics and optimal control. *IFAC Proceedings Volumes*, 38(1):538–543, 2005.
7. Claude Lacoursiere. *Ghosts and machines: regularized variational methods for interactive simulations of multibodies with dry frictional contacts*. PhD thesis, Datavetenskap, 2007.
8. Zachary Manchester and Scott Kuindersma. Variational contact-implicit trajectory optimization. In *International Symposium on Robotics Research (ISRR)*, 2017.
9. Jeongseok Lee, C Karen Liu, Frank C Park, and Siddhartha S Srinivasa. A linear-time variational integrator for multibody systems. In *International Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2016.
10. Michael Posa, Scott Kuindersma, and Russ Tedrake. Optimization and stabilization of trajectories for constrained dynamical systems. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.
11. Ayonga Hereid, Christian M Hubicki, Eric A Cousineau, and Aaron D Ames. Dynamic humanoid locomotion: A scalable formulation for hzd gait optimization. *IEEE Transactions on Robotics*, 2018.
12. Sina Ober-Blöbaum and Nils Saake. Construction and analysis of higher order Galerkin variational integrators. *Advances in Computational Mathematics*, 41(6):955–986, 2015.
13. Sina Ober-Blöbaum. Galerkin variational integrators and modified symplectic Runge–Kutta methods. *IMA Journal of Numerical Analysis*, 37(1):375–406, 2017.
14. Roy Featherstone. *Rigid body dynamics algorithms*. Springer, 2014.
15. Katsu Yamane and Yoshihiko Nakamura. Efficient parallel dynamics computation of human figures. In *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2002.
16. Amir Fijany, Inna Sharf, and Gabriele MT D'Eleuterio. Parallel O (log n) algorithms for computation of manipulator forward dynamics. *IEEE Transactions on Robotics and Automation*, 11(3):389–400, 1995.
17. Taosha Fan, Jarvis Schultz, and Todd D Murphey. Efficient computation of variational integrators in robotic simulation and trajectory optimization: Appendix. https://northwestern.box.com/s/6mgjn3aka1owgem1wt086ashse734385.
18. Justin Carpentier. Analytical derivatives of rigid body dynamics algorithms. In *Robotics: Science and Systems (RSS)*, 2018.
19. Jerry Pratt and Gill Pratt. Intuitive control of a planar bipedal walking robot. In *IEEE International Conference on Robotics and Automation (ICRA)*, 1998.
20. Alexander Shkolnik, Michael Levashov, Ian R Manchester, and Russ Tedrake. Bounding on rough terrain with the LittleDog robot. *The International Journal of Robotics Research*, 30(2):192–215, 2011.
21. Gabe Nelson, Aaron Saunders, Neil Neville, Ben Swilling, Joe Bondaryk, Devin Billings, Chris Lee, Robert Playter, and Marc Raibert. Petman: A humanoid robot for testing chemical protective clothing. *Journal of the Robotics Society of Japan*, 30(4):372–377, 2012.
22. Philip E Gill, Walter Murray, and Michael A Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM review*, 47(1):99–131, 2005.

Efficient Computation of Higher-Order Variational Integrators in Robotic Simulation and Trajectory Optimization: Appendix

Taosha Fan Jarvis Schultz Todd Murphey

Department of Mechanical Engineering, Northwestern University,
2145 Sheridan Road, Evanston, IL 60208, USA
taosha.fan@u.northwestern.edu, jschultz@northwestern.edu,
t-murphey@northwestern.edu

Abstract. This appendix provides the complete $O(n)$ algorithms to compute the Newton direction for higher-order variational integrators and the proofs of the propositions in the paper "Efficient Computation of Higher-Order Variational Integrators in Robotic Simulation and Trajectory Optimization" [1], accepted to the 13th International Workshop on the Algorithmic Foundations of Robotics (WAFR'18). It is assumed that the reader has read the original paper and knows the problem statements and the notation used. The numbering of the equations, algorithms, propositions, etc., is consistent with the numbering used in the original paper.

A Introduction

In the paper "Efficient Computation of Higher-Order Variational Integrators in Robotic Simulation and Trajectory Optimization" [1], we present $O(n)$ algorithms to evaluate the discrete Euler-Lagrange (DEL) equations and compute the Newton direction for solving the DEL equations, and $O(n^2)$ algorithms to linearize the DEL equations. As an appendix to [1], this document provides the complete $O(n)$ algorithms to compute the Newton direction for higher-order variational integrators and the proofs of the propositions in [1], which are not covered in the original paper due to space limitations.

In this appendix, we begin with the complete $O(n)$ algorithms to compute the Newton direction in Section B. In Section C, we give an overview of preliminaries used in the algorithms and proofs. Propositions 1 to 4 in [1, Sections 3 and 4] to compute the higher-order variational integrators are proved in Section D.

For implementation only, the reader only needs to read Algorithms B.1 and B.2 in Section B as well as Algorithms 1 to 3 in [1, Sections 3 and 4]. Sections C and D are not required to read as they present the proofs of the propositions in [1] that do not necessarily aid in implementation.

Even though most of the important content in [1] is reiterated, we still advise the reader to read the original paper to know the problem statements and the notation used. Moreover, as mentioned in the abstract, the numbering of the equations, algorithms, propositions, etc., is consistent with the numbering used in [1]. Therefore, the original paper will not be explicitly cited in the rest of this appendix when we make references to anything in it.

B The $O(n)$ Algorithms to Compute the Newton Direction

In this section, we present Algorithms B.1 and B.2 to compute the Newton direction for higher-order variational integrators. The algorithms are self-contained and we refer the reader to Section C.3 for differentiation on Lie groups that is used to compute $\mathbb{D}_1\overline{F}_i^{k,\alpha}$ in Eq. (B.3b) of Algorithm B.2. The correctness and the $O(n)$ complexity of Algorithms B.1 and B.2 are proved in Section D.2, however, this is not required to read for implementation. We remind the reader that $\delta q_i^{k,\gamma}$ is the Newton direction for $q_i^{k,\gamma}$, and $r_i^{k,\varrho}$ is the residue of the DEL equations Eqs. (7a) and (7b). Moreover, from Proposition 2, Algorithms B.1 and B.2 assume that the inverse of the Jacobian $\mathcal{J}^{-1}(\overline{q}^k)$ exists, and $\overline{F}_i^{k,\alpha}$ and $Q_i^{k,\alpha}$ can be respectively formulated as $\overline{F}_i^{k,\alpha} = \overline{F}_i^{k,\alpha}(g_i^{k,\alpha}, \overline{v}_i^{k,\alpha}, u^{k,\alpha})$ and $Q_i^{k,\alpha} = Q_i^{k,\alpha}(q_i^{k,\alpha}, \dot{q}_i^{k,\alpha}, u^{k,\alpha})$.

There are a number of quantities, such as $D_i^{k,\alpha\rho}$, $\varPhi_i^{k,\alpha\gamma}$, $\zeta_i^{k,\alpha}$, $H_i^{k,\gamma}$, etc., which are recursively introduced in Algorithm B.2 to compute the Newton direction. Since there is no influence on the implementation of the algorithms as long as these quantities are correctly computed, we leave the explanation of their meaning to Section D.2. Similarly, the detailed explanation of $\overline{\eta}_i^{k,\nu}$ and $\overline{\delta}\overline{v}_i^{k,\rho}$ in Algorithm B.1 is left to Sections C.1 and C.2, respectively. For purposes of implementation, the reader only needs to know that these quantities are recursively computed through Algorithms B.1 and B.2.

Algorithm B.1 Recursive Computation of the Newton Direction

initialize $g_0^{k,\alpha} = \mathbf{I}$ and $\overline{v}_0^{k,\alpha} = 0$
for $i = 1 \to n$ **do**
  for $\alpha = 0 \to s$ **do**
    $g_i^{k,\alpha} = g_{\mathrm{par}(i)}^{k,\alpha} g_{\mathrm{par}(i),i}^{k,\alpha}(q_i^{k,\alpha})$
    $\overline{S}_i^{k,\alpha} = \mathrm{Ad}_{g_i^{k,\alpha}} S_i, \quad \overline{M}_i^{k,\alpha} = \mathrm{Ad}_{g_i^{k,\alpha}}^{-T} M_i \mathrm{Ad}_{g_i^{k,\alpha}}^{-1}$
    $\dot{q}_i^{k,\alpha} = \frac{1}{\Delta t}\sum_{\beta=0}^{s} b^{\alpha\beta} q_i^{k,\beta}, \quad \overline{v}_i^{k,\alpha} = \overline{v}_{\mathrm{par}(i)}^{k,\alpha} + \overline{S}_i^{k,\alpha}\cdot \dot{q}_i^{k,\alpha}$
    $\dot{\overline{S}}_i^{k,\alpha} = \mathrm{ad}_{\overline{v}_i^{k,\alpha}} \overline{S}_i^{k,\alpha}$
  end for
end for
for $i = n \to 1$ **do**
  use Algorithm B.2 to evaluate
    a) $D_i^{k,\alpha\rho}$, $G_i^{k,\alpha\nu}$, $l_i^{k,\alpha}$ and $\overline{\mu}_i^{k,\alpha}$
    b) $\varPi_i^{k,\alpha\rho}$, $\varPsi_i^{k,\alpha\nu}$, $\zeta_i^{k,\alpha}$ and $\overline{\varGamma}_i^{k,\alpha}$
    c) $H_i^{k,\alpha}$ and $\varPhi_i^{k,\alpha}$
    d) $X_i^{k,\alpha\rho}$, $Y_i^{k,\alpha\nu}$ and $y_i^{k,\alpha}$
end for
initialize $\overline{\eta}_0^{k,\nu} = 0$ and $\overline{\delta}\overline{v}_0^{k,\rho} = 0$
for $i = 1 \to n$ **do**

for $\gamma = 1 \to s$ **do**

$\delta q_i^{k,\gamma} = \sum\limits_{\rho=0}^{s} X_i^{k,\gamma\rho} \cdot \overline{\delta v}_{\mathrm{par}(i)}^{k,\rho} + \sum\limits_{\nu=1}^{s} Y_i^{k,\gamma\nu} \cdot \overline{\eta}_{\mathrm{par}(i)}^{k,\nu} + y_i^{k,\gamma}$

end for

for $\nu = 1 \to s$ **do**

$\overline{\eta}_i^{k,\nu} = \overline{\eta}_{\mathrm{par}(i)}^{k,\nu} + \overline{S}_i^{k,\nu} \cdot \delta q_i^{k,\nu}$

end for

for $\rho = 0 \to s$ **do**

$\delta \dot{q}_i^{k,\rho} = \frac{1}{\Delta t} \sum\limits_{\gamma=1}^{s} b^{\rho\gamma} \cdot \delta q_i^{k,\gamma}$

$\overline{\delta v}_i^{k,\rho} = \overline{\delta v}_{\mathrm{par}(i)}^{k,\rho} + \dot{\overline{S}}_i^{k,\rho} \cdot \delta q_i^{k,\rho} + \overline{S}_i^{k,\rho} \cdot \delta \dot{q}_i^{k,\rho}$

end for

end for

Algorithm B.2 Recursive Computation of the Newton Direction – Backward Pass

1: $\forall \alpha = 0, 1, \cdots, s, \forall \rho = 0, 1, \cdots, s$ and $\forall \nu = 0, 1, \cdots, s-1$,

$$D_i^{k,\alpha\rho} = \sigma^{\alpha\rho} \overline{M}_i^{k,\alpha} + \sum_{j \in \mathrm{chd}(i)} \Big(D_j^{k,\alpha\rho} + \sum_{\gamma=1}^{s} H_j^{k,\alpha\gamma} X_j^{k,\gamma\rho} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}} \overline{S}_j^{k,\alpha} X_j^{k,\alpha\rho} \Big), \tag{B.1a}$$

$$G_i^{k,\alpha\nu} = \sum_{j \in \mathrm{chd}(i)} \Big(G_j^{k,\alpha\nu} + \sum_{\gamma=1}^{s} H_j^{k,\alpha\gamma} Y_j^{k,\gamma\nu} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}} \overline{S}_j^{k,\alpha} Y_j^{k,\alpha\nu} \Big), \tag{B.1b}$$

$$l_i^{k,\alpha} = \sum_{j \in \mathrm{chd}(i)} \Big(l_j^{k,\alpha} + \sum_{\gamma=1}^{s} H_j^{k,\alpha\gamma} y_j^{k,\gamma} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}} \overline{S}_j^{k,\alpha} y_j^{k,\alpha} \Big), \tag{B.1c}$$

$$\overline{\mu}_i^{k,\alpha} = \overline{M}_i^{k,\alpha} \overline{v}_i^{k,\alpha} + \sum_{j \in \mathrm{chd}(i)} \mu_j^{k,\alpha}$$

in which

$$\sigma^{\alpha\rho} = \begin{cases} 1 & \alpha = \rho, \\ 0 & \alpha \neq \rho \end{cases} \qquad \text{and} \qquad \overline{\sigma}^{\alpha 0} = \begin{cases} 1 & \alpha \neq 0, \\ 0 & \alpha = 0 \end{cases} \tag{B.2}$$

2: $\forall \alpha = 0, 1, \cdots, s-1, \forall \rho = 0, 1, \cdots, s$ and $\forall \nu = 0, 1, \cdots, s-1$,

$$\varPi_i^{k,\alpha\rho} = \sigma^{\alpha\rho} \mathbb{D}_2 \overline{F}_i^{k,\alpha} + \sum_{j \in \mathrm{chd}(i)} \Big(\varPi_j^{k,\alpha\rho} + \sum_{\gamma=1}^{s} \varPhi_j^{k,\alpha\gamma} X_j^{k,\gamma\rho} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\overline{\varGamma}_j^{k,\alpha}} \overline{S}_j^{k,\alpha} X_j^{k,\alpha\rho} \Big), \tag{B.3a}$$

$$
\begin{aligned}
\Psi_i^{k,\alpha\nu} = \sigma^{\alpha\nu}\Big(\mathbb{D}_1\overline{F}_i^{k,\alpha} + \mathrm{ad}^D_{\overline{F}_i^{k,\alpha}} - \mathbb{D}_2\overline{F}_i^{k,\alpha}\mathrm{ad}_{\overline{v}_i^{k,\alpha}}\Big) + \\
\sum_{j\in\mathrm{chd}(i)}\Big(\Psi_j^{k,\alpha\nu} + \sum_{\gamma=1}^{s}\Phi_j^{k,\alpha\gamma}Y_j^{k,\gamma\nu} - \overline{\sigma}^{\alpha 0}\mathrm{ad}^D_{\overline{\Gamma}_j^{k,\alpha}}\overline{S}_j^{k,\alpha}Y_j^{k,\alpha\nu}\Big),
\end{aligned}
\tag{B.3b}
$$

$$
\zeta_i^{k,\alpha} = \sum_{j\in\mathrm{chd}(i)}\Big(\zeta_j^{k,\alpha} + \sum_{\gamma=1}^{s}\Phi_j^{k,\alpha\gamma}y_j^{k,\gamma} - \overline{\sigma}^{\alpha 0}\mathrm{ad}^D_{\overline{\Gamma}_j^{k,\alpha}}\overline{S}_j^{k,\alpha}y_j^{k,\alpha}\Big),
\tag{B.3c}
$$

$$
\overline{\Gamma}_i^{k,\alpha} = \overline{F}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)}\overline{\Gamma}_j^{k,\alpha}
$$

3: $\forall\alpha = 0, 1, \cdots, s$ and $\forall\gamma = 1, 2, \cdots, s$,

$$
H_i^{k,\alpha\gamma} = D_i^{k,\alpha\gamma}\dot{\overline{S}}_i^{k,\gamma} + G_i^{k,\alpha\gamma}\overline{S}_i^{k,\gamma} + \frac{1}{\Delta t}\sum_{\rho=0}^{s}b^{\rho\gamma}D_i^{k,\alpha\rho}\overline{S}_i^{k,\rho}.
$$

4: $\forall\alpha = 0, 1, \cdots, s-1$ and $\forall\gamma = 1, 2, \cdots, s$,

$$
\Phi_i^{k,\alpha\gamma} = \Pi_i^{k,\alpha\gamma}\dot{\overline{S}}_i^{k,\gamma} + \Psi_i^{k,\alpha\gamma}\overline{S}_i^{k,\gamma} + \frac{1}{\Delta t}\sum_{\rho=0}^{s}b^{\rho\gamma}\Pi_i^{k,\alpha\rho}\overline{S}_i^{k,\rho}.
$$

5: $\forall\alpha = 0, 1, \cdots, s-1, \forall\rho = 0, 1, \cdots, s$ and $\forall\nu = 0, 1, \cdots, s-1$,

$$
\begin{aligned}
\Theta_i^{k,\alpha\rho} &= w^\alpha\Delta t\cdot\big(\dot{\overline{S}}_i^{k,\alpha^T}D_i^{k,\alpha\rho} + \sigma^{\alpha\rho}\overline{S}_i^{k,\alpha^T}\mathrm{ad}^D_{\mu_i^{k,\alpha}}\big) + \overline{S}_i^{k,\alpha^T}\Pi_i^{k,\alpha\rho}, \\
\Xi_i^{k,\alpha\nu} &= w^\alpha\Delta t\cdot\dot{\overline{S}}_i^{k,\alpha^T}G_i^{k,\alpha\nu} + \overline{S}_i^{k,\alpha^T}\Psi_i^{k,\alpha\nu}.
\end{aligned}
$$

6: $\forall\alpha = 0, 1, \cdots, s-1, \forall\rho = 0, 1, \cdots, s$ and $\forall\nu = 0, 1, \cdots, s-1$,

$$
\begin{aligned}
\overline{\Theta}_i^{k,\alpha\rho} &= \Theta_i^{k,\alpha\rho} + \sum_{\beta=0}^{s}a^{\alpha\beta}\overline{S}_i^{k,\beta^T}D_i^{k,\beta\rho}, \\
\overline{\Xi}_i^{k,\alpha\nu} &= \Xi_i^{k,\alpha\nu} + \sum_{\beta=0}^{s}a^{\alpha\beta}\overline{S}_i^{k,\beta^T}G_i^{k,\beta\nu}, \\
\overline{\xi}_i^{k,\alpha} &= w^\alpha\Delta t\cdot\dot{\overline{S}}_i^{k,\alpha^T}l_i^{k,\alpha} + \overline{S}_i^{k,\alpha^T}\zeta_i^{k,\alpha} + \sum_{\beta=0}^{s}a^{\alpha\beta}\overline{S}_i^{k,\beta^T}l_i^{k,\beta}.
\end{aligned}
$$

7: $\forall\alpha = 0, 1, \cdots, s-1$ and $\forall\gamma = 1, 2, \cdots, s$,

$$
\begin{aligned}
\Lambda_i^{k,\alpha\gamma} =& w^\alpha\Delta t\cdot\dot{\overline{S}}_i^{k,\alpha^T}H_i^{k,\alpha\gamma} + \overline{S}_i^{k,\alpha^T}\Phi_i^{k,\alpha\gamma} + \sum_{\beta=0}^{s}a^{\alpha\beta}\overline{S}_i^{k,\beta^T}H_i^{k,\beta\gamma} + \\
&\sigma^{\alpha\gamma}\left(\mathbb{D}_1Q_i^{k,\alpha} + w^\alpha\Delta t\cdot\overline{S}_i^{k,\alpha^T}\mathrm{ad}^D_{\mu_i^{k,\alpha}}\dot{\overline{S}}_i^{k,\alpha}\right) + \frac{1}{\Delta t}b^{\alpha\gamma}\cdot\mathbb{D}_2Q_i^{k,\alpha}
\end{aligned}
$$

with which $\Lambda_i^k = \left[\Lambda_i^{k,\alpha\gamma}\right] \in \mathbb{R}^{s\times s}$

8: $\forall \gamma = 1, 2, \cdots, s$ and $\forall \varrho = 0, 1, \cdots, s-1$, compute $\overline{\Lambda}_i^{k,\gamma\varrho}$ such that ${\Lambda_i^k}^{-1} = \left[\overline{\Lambda}_i^{k,\gamma\varrho}\right] \in \mathbb{R}^{s\times s}$

9: $\forall \gamma = 1, 2, \cdots, s$, $\forall \rho = 0, 1, \cdots, s$ and $\forall \nu = 1, 2, \cdots, s$

$$X_i^{k,\gamma\rho} = -\sum_{\varrho=0}^{s-1} \overline{\Lambda}_i^{k,\gamma\varrho} \cdot \overline{\Theta}_i^{k,\varrho\rho},$$

$$Y_i^{k,\gamma\nu} = -\sum_{\varrho=0}^{s-1} \overline{\Lambda}_i^{k,\gamma\varrho} \cdot \overline{\Xi}_i^{k,\varrho\nu},$$

$$y_i^{k,\gamma} = -\sum_{\varrho=0}^{s-1} \overline{\Lambda}_i^{k,\gamma\varrho} \left(r_i^{k,\varrho} + \overline{\xi}_i^{k,\varrho}\right)$$

C Preliminaries

In this section, we present additional preliminaries used in Algorithms B.1 and B.2 and the proofs of Propositions 1 to 4. In Section C.1, we extend the contents of Section 2.3 for the computation of variations and derivatives. In Sections C.2 and C.3, we respectively introduce the notion of the spatial variation for spatial quantities and the differentiation on Lie groups, which are mainly used in Algorithms B.1 and B.2 and the proof of Proposition 2.

C.1 The Tree Representation Revisited

In addition to the computation of rigid body dynamics as those in Section 2.3, the tree representation can also be used to compute the variations and derivatives.

As is known, in the tree representation, the configuration $g_i \in SE(3)$ of rigid body i is

$$g_i = g_{\mathrm{par}(i)} g_{\mathrm{par}(i),i}(q_i) \tag{C.1}$$

in which $g_{\mathrm{par}(i),i}(q_i) = g_{\mathrm{par}(i),i}(0)\exp(\hat{S}_i q_i)$ and S_i is the body Jacobian of joint i with respect to frame $\{i\}$. In addition, the spatial Jacobian of joint i with respect to frame $\{0\}$ is

$$\overline{S}_i = \mathrm{Ad}_{g_i} S_i \tag{C.2}$$

in which S_i is constant by definition. Using Eqs. (C.1) and (C.2) as well as $\mathrm{Ad}_{g_i} S_i = \left(g_i \hat{S}_i g_i^{-1}\right)^\vee$, we obtain $\overline{\eta}_i = (\delta g_i g_i^{-1})^\vee$ as

$$\overline{\eta}_i = \overline{\eta}_{\mathrm{par}(i)} + \overline{S}_i \cdot \delta q_i, \tag{C.3}$$

or equivalently,

$$\overline{\eta}_i = \overline{S}_i \cdot \delta q_i + \sum_{j\in\mathrm{anc}(i)}^{n} \overline{S}_j \cdot \delta q_j \tag{C.4}$$

and furthermore,

$$\left(\frac{\partial g_i}{\partial q_j} g_i^{-1}\right)^{\vee} = \begin{cases} \overline{S}_j & j \in \text{anc}(i) \cup \{i\}, \\ 0 & \text{otherwise}, \end{cases} \tag{C.5a}$$

$$\left(\frac{\partial g_j}{\partial q_i} g_i^{-1}\right)^{\vee} = \begin{cases} \overline{S}_i & j \in \text{des}(i) \cup \{i\}, \\ 0 & \text{otherwise}. \end{cases} \tag{C.5b}$$

In addition, from Eqs. (C.2) and (C.3), $\delta \mathrm{Ad}_{g_i} = \mathrm{ad}_{\overline{\eta}_i} \mathrm{Ad}_{g_i}$ and $\mathrm{ad}_{\overline{S}_i} \overline{S}_i = 0$, we obtain

$$\delta \overline{S}_i = \mathrm{ad}_{\overline{\eta}_i} \overline{S}_i = -\mathrm{ad}_{\overline{S}_i} \overline{\eta}_i = \mathrm{ad}_{\overline{\eta}_{\mathrm{par}(i)}} \overline{S}_i = -\mathrm{ad}_{\overline{S}_i} \overline{\eta}_{\mathrm{par}(i)}. \tag{C.6}$$

Moreover, as a result of Eqs. (C.4) to (C.6), we further obtain

$$\frac{\partial \overline{S}_i}{\partial q_j} = \begin{cases} \mathrm{ad}_{\overline{S}_j} \overline{S}_i & j \in \text{anc}(i), \\ 0 & \text{otherwise}, \end{cases} \tag{C.7a}$$

$$\frac{\partial \overline{S}_j}{\partial q_i} = \begin{cases} \mathrm{ad}_{\overline{S}_i} \overline{S}_j & j \in \text{des}(i), \\ 0 & \text{otherwise}. \end{cases} \tag{C.7b}$$

Since the spatial velocity $\overline{v}_i$ of rigid body i is

$$\begin{aligned} \overline{v}_i &= \overline{S}_i \cdot \dot{q}_i + \sum_{j \in \text{anc}(i)} \overline{S}_j \cdot \dot{q}_j \\ &= \overline{v}_{\mathrm{par}(i)} + \overline{S}_i \cdot \dot{q}_i, \end{aligned} \tag{C.8}$$

we obtain

$$\begin{aligned} \delta \overline{v}_i &= \delta \overline{S}_i \cdot \dot{q}_i + \overline{S}_i \cdot \delta \dot{q}_i + \sum_{j \in \text{anc}(i)} \left(\delta \overline{S}_j \cdot \dot{q}_j + \overline{S}_j \cdot \delta \dot{q}_j \right) \\ &= \delta \overline{v}_{\mathrm{par}(i)} + \delta \overline{S}_i \cdot \dot{q}_i + \overline{S}_i \cdot \delta \dot{q}_i. \end{aligned}$$

Substitute Eq. (C.6) into the equation above, the result is

$$\begin{aligned} \delta \overline{v}_i &= \mathrm{ad}_{\overline{\eta}_i} \overline{S}_i \cdot \dot{q}_i + \overline{S}_i \cdot \delta \dot{q}_i + \sum_{j \in \text{anc}(i)} \left(\mathrm{ad}_{\overline{\eta}_j} \overline{S}_j \cdot \dot{q}_j + \overline{S}_j \cdot \delta \dot{q}_j \right) \\ &= \delta \overline{v}_{\mathrm{par}(i)} + \mathrm{ad}_{\overline{\eta}_i} \overline{S}_i \cdot \dot{q}_i + \overline{S}_i \cdot \delta \dot{q}_i. \end{aligned} \tag{C.9}$$

From Eqs. (C.6) to (C.9), we obtain

$$\frac{\partial \overline{v}_i}{\partial \dot{q}_j} = \begin{cases} S_j & j \in \text{anc}(i) \cup \{i\}, \\ 0 & \text{otherwise}, \end{cases} \tag{C.10a}$$

$$\frac{\partial \overline{v}_j}{\partial \dot{q}_i} = \begin{cases} S_i & j \in \text{des}(i) \cup \{i\}, \\ 0 & \text{otherwise}, \end{cases} \tag{C.10b}$$

and

$$\frac{\partial \overline{v}_i}{\partial q_j} = \begin{cases} \mathrm{ad}_{\overline{S}_j} (\overline{v}_i - \overline{v}_j) & j \in \text{anc}(i) \cup \{i\}, \\ 0 & \text{otherwise}, \end{cases} \tag{C.11a}$$

$$\frac{\partial \overline{v}_j}{\partial q_i} = \begin{cases} \mathrm{ad}_{\overline{S}_i}(\overline{v}_j - \overline{v}_i) & j \in \mathrm{des}(i) \cup \{i\}, \\ 0 & \text{otherwise.} \end{cases} \tag{C.11b}$$

In addition, from Eqs. (C.2) and (C.8), $\mathrm{Ad}_{\dot{g}_i} = \mathrm{ad}_{\overline{v}_i}\mathrm{Ad}_{g_i}$ and $\mathrm{ad}_{\overline{S}_i}\overline{S}_i = 0$, we obtain

$$\dot{\overline{S}}_i = \mathrm{ad}_{\overline{v}_i}\overline{S}_i = -\mathrm{ad}_{\overline{S}_i}\overline{v}_i = \mathrm{ad}_{\overline{v}_{\mathrm{par}(i)}}\overline{S}_i = -\mathrm{ad}_{\overline{S}_i}\overline{v}_{\mathrm{par}(i)}. \tag{C.12}$$

As for the spatial inertia matrix $\overline{M}_i = \mathrm{Ad}_{g_i}^{-T} M_i \mathrm{Ad}_{g_i}^{-1}$, algebraic manipulation shows that

$$\delta\overline{M}_i = -\mathrm{ad}_{\overline{\eta}_i}^T \cdot \overline{M}_i - \overline{M}_i \cdot \mathrm{ad}_{\overline{\eta}_i}, \tag{C.13}$$

and from Eqs. (C.3) to (C.5) and Eq. (C.13), we obtain

$$\frac{\partial \overline{M}_i}{\partial q_j} = \begin{cases} -\mathrm{ad}_{\overline{S}_j}^T \overline{M}_i - \overline{M}_i \mathrm{ad}_{\overline{S}_j} & j \in \mathrm{anc}(i) \cup \{i\}, \\ 0 & \text{otherwise,} \end{cases} \tag{C.14a}$$

$$\frac{\partial \overline{M}_j}{\partial q_i} = \begin{cases} -\mathrm{ad}_{\overline{S}_i}^T \overline{M}_j - \overline{M}_j \mathrm{ad}_{\overline{S}_i} & j \in \mathrm{des}(i) \cup \{i\}, \\ 0 & \text{otherwise.} \end{cases} \tag{C.14b}$$

In Sections D.1 to D.4, Eq. (C.3) to (C.14) will be used to prove Propositions 1 to 4.

C.2 The Spatial Variation

In this subsection, we introduce the *spatial variation* $\overline{\delta}\,\overline{(\cdot)}$ that is used in Algorithms B.1 and B.2 and the proof of Proposition 2. Note that the notion of the spatial variation $\overline{\delta}\,\overline{(\cdot)}$ only applies to the spatial quantities $\overline{(\cdot)}$ of $T_eSE(3)$ or $T_e^*SE(3)$ that are described in the spatial frame.

If $\overline{a}, a \in T_eSE(3)$ are related as $\overline{a} = \mathrm{Ad}_g a$ in which $g \in SE(3)$, we have

$$\delta\overline{a} = \mathrm{Ad}_g \delta a + \mathrm{ad}_{\overline{\eta}}\overline{a}$$

in which $\overline{\eta} = (\delta g g^{-1})^\vee$. For numerical simplicity, it is sometimes preferable to have the variations of $\overline{a}$ and a still related by Ad_g. Therefore, we define the spatial variation $\overline{\delta}\overline{a}$ to be

$$\overline{\delta}\overline{a} = \delta\overline{a} - \mathrm{ad}_{\overline{\eta}}\overline{a} \tag{C.15}$$

such that $\overline{\delta}\overline{a} = \mathrm{Ad}_g \delta a$ as long as $\overline{a} = \mathrm{Ad}_g a$. In a similar way, if $\overline{b}^*, b^* \in T_e^*SE(3)$ are related as $\overline{b}^* = \mathrm{Ad}_g^{-T} b^*$, we obtain

$$\delta\overline{b}^* = \mathrm{Ad}_g^{-T}\delta b^* - \mathrm{ad}_{\overline{\eta}}^T \overline{b}^*.$$

Similar to Eq. (C.15), the spatial variation $\overline{\delta}\,\overline{b}^*$ is defined to be

$$\overline{\delta}\,\overline{b}^* = \delta\overline{b}^* + \mathrm{ad}_{\overline{\eta}}^T \overline{b}^* \tag{C.16}$$

such that $\overline{\delta}\,\overline{b}^* = \mathrm{Ad}_g^{-T}\delta b^*$ as long as $\overline{b}^* = \mathrm{Ad}_g^{-T} b^*$. In addition, note that $\delta\big(b^{*T} a\big) = \delta b^{*T} a + b^{*T}\delta a = \overline{\delta}\,\overline{b}^{*T}\overline{a} + \overline{b}^{*T}\overline{\delta}\overline{a}$ and $\delta(\overline{b}^{*T}\overline{a}) = \delta(b^{*T}a)$, we have

$$\delta(\overline{b}^{*T}\overline{a}) = \overline{\delta}\,\overline{b}^{*T}\overline{a} + \overline{b}^{*T}\overline{\delta}\overline{a}. \tag{C.17}$$

In general, the spatial variations $\overline{\delta}\,\overline{(\cdot)}$ are the infinitesimal changes of spatial quantities in either the Lie algebra $T_e SE(3)$ or the dual Lie algebra $T_e^* SE(3)$ after canceling out the influences of the frame change.

In Section 3, we have a number of spatial quantities that are defined in $T_e SE(3)$ and $T_e^* SE(3)$, whose spatial variations $\overline{\delta}\,\overline{(\cdot)}$ can be computed in the tree representation.

Following Eqs. (C.2), (C.6) and (C.15), for $\overline{S}_i^{k,\alpha} = \mathrm{Ad}_{g_i^{k,\alpha}} S_i$, the spatial variation $\overline{\delta}\,\overline{S}_i^{k,\alpha}$ is

$$\overline{\delta}\,\overline{S}_i^{k,\alpha} = 0 \tag{C.18}$$

though $\delta \overline{S}_i^{k,\alpha} = \mathrm{ad}_{\overline{\eta}_i^{k,\alpha}} \overline{S}_i^{k,\alpha}$ is usually not zero. In addition, according to Eqs. (C.9) and (C.15), we have

$$\overline{\delta}\overline{v}_i^{k,\alpha} = \delta\overline{v}_{\mathrm{par}(i)}^{k,\alpha} + \mathrm{ad}_{\overline{\eta}_i^{k,\alpha}} \overline{S}_i^{k,\alpha}\cdot \dot{q}_i^{k,\alpha} + \overline{S}_i^{k,\alpha}\cdot\delta\dot{q}_i^{k,\alpha} - \mathrm{ad}_{\overline{\eta}_i^{k,\alpha}}\overline{v}_i^{k,\alpha}$$

Substitute Eqs. (C.3) and (C.8) into the equation above to expand $\mathrm{ad}_{\overline{\eta}_i^{k,\alpha}}\overline{v}_i^{k,\alpha}$ and apply Eqs. (C.6) and (C.12), it can be shown that

$$\overline{\delta}\overline{v}_i^{k,\alpha} = \overline{\delta}\overline{v}_{\mathrm{par}(i)}^{k,\alpha} + \dot{\overline{S}}_i^{k,\alpha}\cdot\delta q_i^{k,\alpha} + \overline{S}_i^{k,\alpha}\cdot\delta\dot{q}^{k,\alpha}. \tag{C.19}$$

In terms of $\overline{\mu}_i^{k,\alpha}$, $\overline{\Gamma}_i^{k,\alpha}$ and $\overline{\Omega}_i^{k,\alpha}$ in Eq. (7), which are spatial quantities in $T_e^* SE(3)$, we can still implement the tree representation to compute the spatial variation. According to Definition 1, we have

$$\delta\overline{\mu}_i^{k,\alpha} = \delta(\overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha}) + \sum_{j\in\mathrm{chd}(i)}\delta\overline{\mu}_j^{k,\alpha}.$$

From Eq. (C.16), the spatial variation $\overline{\delta}\overline{\mu}_i^{k,\alpha}$ is

$$\overline{\delta}\overline{\mu}_i^{k,\alpha} = \delta(\overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha}) + \sum_{j\in\mathrm{chd}(i)}\delta\overline{\mu}_j^{k,\alpha} + \mathrm{ad}^T_{\overline{\eta}_i^{k,\alpha}}\overline{\mu}_i^{k,\alpha}.$$

Using $\overline{\mu}_i^{k,\alpha} = \overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)}\overline{\mu}_j^{k,\alpha}$ and $\overline{\eta}_i^{k,\alpha} = \overline{\eta}_j^{k,\alpha} - \overline{S}_j^{k,\alpha}\cdot\delta q_j^{k,\alpha}$, we have

$$\begin{aligned}\overline{\delta}\overline{\mu}_i^{k,\alpha} = \delta(\overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha}) + \mathrm{ad}^T_{\overline{\eta}_i^{k,\alpha}}(\overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha}) + \\ \sum_{j\in\mathrm{chd}(i)}\left(\delta\overline{\mu}_j^{k,\alpha} + \mathrm{ad}^T_{\overline{\eta}_j^{k,\alpha}}\overline{\mu}_j^{k,\alpha} - \mathrm{ad}^T_{\overline{S}_j^{k,\alpha}}\overline{\mu}_j^{k,\alpha}\cdot\delta q_j^{k,\alpha}\right)\end{aligned} \tag{C.20}$$

As a result of Eqs. (C.13) and (C.15), $\delta(\overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha}) + \mathrm{ad}^T_{\overline{\eta}_i^{k,\alpha}}(\overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha})$ is

$$\begin{aligned} \delta(\overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha}) + \mathrm{ad}^T_{\overline{\eta}_i^{k,\alpha}}(\overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha}) &= \overline{M}_i^{k,\alpha}(\delta\overline{v}_i^{k,\alpha} - \mathrm{ad}_{\overline{\eta}_i^{k,\alpha}}\overline{v}_i^{k,\alpha}) \\ &= \overline{M}_i^{k,\alpha}\overline{\delta}\overline{v}_i^{k,\alpha}. \end{aligned} \tag{C.21}$$

From Eqs. (C.16) and (C.21) and $\mathrm{ad}^T_{\overline{S}_j^{k,\alpha}}\overline{\mu}_j^{k,\alpha} = \mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}}\overline{S}_j^{k,\alpha}$, Eq. (C.20) is simplified to

$$\begin{aligned} \overline{\delta}\overline{\mu}_i^{k,\alpha} &= \overline{M}_i^{k,\alpha}\overline{\delta}\overline{v}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)}\left(\overline{\delta}\overline{\mu}_j^{k,\alpha} - \mathrm{ad}^T_{\overline{S}_j^{k,\alpha}}\overline{\mu}_j^{k,\alpha}\cdot\delta q_j^{k,\alpha}\right) \\ &= \overline{M}_i^{k,\alpha}\overline{\delta}\overline{v}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)}\left(\overline{\delta}\overline{\mu}_j^{k,\alpha} - \mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}}\overline{S}_j^{k,\alpha}\cdot\delta q_j^{k,\alpha}\right). \end{aligned} \tag{C.22}$$

In a similar way, for the spatial variation $\overline{\delta}\overline{\varGamma}_i^{k,\alpha}$, we obtain

$$\begin{aligned} \overline{\delta}\overline{\varGamma}_i^{k,\alpha} &= \overline{\delta}\overline{F}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)}\left(\overline{\delta}\overline{\varGamma}_j^{k,\alpha} - \mathrm{ad}^T_{\overline{S}_j^{k,\alpha}}\overline{\varGamma}_j^{k,\alpha}\cdot\delta q_j^{k,\alpha}\right) \\ &= \overline{\delta}\overline{F}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)}\left(\overline{\delta}\overline{\varGamma}_j^{k,\alpha} - \mathrm{ad}^D_{\overline{\varGamma}_j^{k,\alpha}}\overline{S}_j^{k,\alpha}\cdot\delta q_j^{k,\alpha}\right). \end{aligned} \tag{C.23}$$

As for $\overline{\varOmega}_i^{k,\alpha} = w^\alpha\varDelta t\cdot\mathrm{ad}^T_{\overline{v}_i^{k,\alpha}}\cdot\overline{\mu}_i^{k,\alpha} + \overline{\varGamma}_i^{k,\alpha}$, from Eqs. (C.15) and (C.16), algebraic manipulation shows that

$$\begin{aligned} \overline{\delta}\overline{\varOmega}_i^{k,\alpha} &= \delta\overline{\varOmega}_i^{k,\alpha} + \mathrm{ad}^T_{\overline{\eta}_i^{k,\alpha}}\overline{\varOmega}_i^{k,\alpha} \\ &= w^\alpha\varDelta t\cdot\left(\mathrm{ad}^T_{\overline{v}_i^{k,\alpha}}\cdot\overline{\delta}\overline{\mu}_i^{k,\alpha} + \mathrm{ad}^T_{\overline{\delta}\overline{v}_i^{k,\alpha}}\overline{\mu}_i^{k,\alpha}\right) + \overline{\delta}\overline{\varGamma}_i^{k,\alpha} \\ &= w^\alpha\varDelta t\cdot\left(\mathrm{ad}^T_{\overline{v}_i^{k,\alpha}}\cdot\overline{\delta}\overline{\mu}_i^{k,\alpha} + \mathrm{ad}^D_{\overline{\mu}_i^{k,\alpha}}\overline{\delta}\overline{v}_i^{k,\alpha}\right) + \overline{\delta}\overline{\varGamma}_i^{k,\alpha}. \end{aligned} \tag{C.24}$$

In Section D.2, Eqs. (C.18), (C.19) and (C.22) to (C.24) will be used to prove Proposition 2.

C.3 Differentiation on Lie Groups

For an analytical function $f:\mathbb{R}^n\to\mathbb{R}$, the directional derivative at $x\in\mathbb{R}^n$ in the direction δx is defined to be

$$\mathbb{D}f(x)\cdot\delta x = \frac{\mathrm{d}}{\mathrm{d}t}f(x+t\cdot\delta x)\bigg|_{t=0}$$

in which $\mathbb{D}f(x) = \begin{bmatrix}\frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \cdots & \frac{\partial f}{\partial x_n}\end{bmatrix}^T \in\mathbb{R}^n$.

In a similar way, we might define the directional derivative on Lie groups using the Lie algebra and the exponential map as follows.

Definition C.1. *If G is a n-dimensional smooth Lie group and $f : G \longrightarrow \mathbb{R}$ is a smooth function on G, the directional derivative at $g \in G$ in the direction $\overline{\eta} = \delta g g^{-1} \in T_e G$ is defined to be*

$$\mathbb{D}f(g) \cdot \overline{\eta} = \frac{\mathrm{d}}{\mathrm{d}t} f\left(\exp\left(t \cdot \overline{\eta}\right) g\right)\Big|_{t=0}.$$

Moreover, if $\overline{e}_1$, $\overline{e}_2$, $\cdots$, $\overline{e}_n$ is a basis for the Lie algebra $T_e G$, then $\mathbb{D}f(g)$ can be explicitly written as

$$\mathbb{D}f(g) = \frac{\mathrm{d}}{\mathrm{d}t}\left[f\left(\exp\left(t \cdot \overline{e}_1\right) g\right) \;\; f\left(\exp\left(t \cdot \overline{e}_2\right) g\right) \;\; \cdots \;\; f\left(\exp\left(t \cdot \overline{e}_n\right) g\right)\right]^T\Big|_{t=0}.$$

In regard to Lie group theory, $\mathbb{R}^n$ is also a smooth Lie group for which the binary operation is addition, the Lie algebra is itself and the exponential map is the identity map. Furthermore, the definition of directional derivatives on Lie groups in Definition C.1 is consistent with the definition of directional derivatives in $\mathbb{R}^n$. Therefore, it is without loss of any generality to interpret all the quantities in this paper as elements of Lie groups and all the derivatives in this paper as derivatives on Lie groups that are defined by Definition C.1.

In this paper, following the notation in multivariate calculus, if $f : G_1 \times G_2 \times \cdots \times G_d \to \mathbb{R}$ is a smooth function in which $G_1, G_2, \cdots, G_d$ are Lie groups, we use $\mathbb{D}_i f$ to denote the derivative with respect to G_i. In particular, for $\overline{F}_i^{k,\alpha} = \overline{F}_i^{k,\alpha}(g_i^{k,\alpha}, \overline{v}_i^{k,\alpha}, u_i^{k,\alpha})$ that is used for the computation of the Newton direction in Algorithm B.2, note that $\mathbb{D}_1 \overline{F}_i^{k,\alpha}$ is the derivative with respect to $g_i^{k,\alpha}$ and $\mathbb{D}_2 \overline{F}_i^{k,\alpha}$ is the derivative with respect to $\overline{v}_i^{k,\alpha}$.

D Proof of Propositions

In this section, we review and prove Propositions 1 to 4 in [1] though these proofs are not necessary for implementation.

D.1 Proof of Proposition 1

In Section 3.1, we define the discrete articulated body momentum and discrete articulated body impulse are respectively as follows.

Definition 1. *The discrete articulated body momentum $\overline{\mu}_i^{k,\alpha} \in \mathbb{R}^6$ for articulated body i is defined to be*

$$\overline{\mu}_i^{k,\alpha} = \overline{M}_i^{k,\alpha} \overline{v}_i^{k,\alpha} + \sum_{j \in \mathrm{chd}(i)} \overline{\mu}_j^{k,\alpha} \qquad \forall \alpha = 0, 1, \cdots, s \tag{D.1}$$

in which $\overline{M}_i^{k,\alpha}$ and $\overline{v}_i^{k,\alpha}$ are respectively the spatial inertia matrix and spatial velocity of rigid body i.

Definition 2. *Suppose $\overline{F}_i(t) \in \mathbb{R}^6$ is the sum of all the wrenches directly acting on rigid body i, which does not include those applied or transmitted through the joints that are connected to rigid body i. The discrete articulated body impulse $\overline{\Gamma}_i^{k,\alpha} \in \mathbb{R}^6$ for articulated body i is defined to be*

$$\overline{\Gamma}_i^{k,\alpha} = \overline{F}_i^{k,\alpha} + \sum_{j\in \mathrm{chd}(i)} \overline{\Gamma}_j^{k,\alpha} \tag{D.2}$$

in which $\overline{F}_i^{k,\alpha} = \omega^\alpha \overline{F}_i(t^{k,\alpha})\Delta t \in \mathbb{R}^6$ is the discrete impulse acting on rigid body i. Note that $\overline{F}_i(t)$, $\overline{F}_i^{k,\alpha}$ and $\overline{\Gamma}_i^{k,\alpha}$ are expressed in frame $\{0\}$.

The DEL equations Eq. (5) can be recursively evaluated with $\overline{\mu}_i^{k,\alpha}$ and $\overline{F}_i^{k,\alpha}$ as Proposition 1 indicates.

Proposition 1. *If $Q_i(t) \in \mathbb{R}$ is the sum of all joint forces applied to joint i and $p^k = \left[p_1^k\ p_2^k \cdots p_n^k\right]^T \in \mathbb{R}^n$ is the discrete momentum, the DEL equations Eq.* (5) *can be evaluated as*

$$r_i^{k,0} = p_i^k + \overline{S}_i^{k,0^T} \cdot \overline{\Omega}_i^{k,0} + \sum_{\beta=0}^{s} a^{0\beta} \overline{S}_i^{k,\beta^T} \cdot \overline{\mu}_i^{k,\beta} + Q_i^{k,0}, \tag{D.3a}$$

$$r_i^{k,\alpha} = \overline{S}_i^{k,\alpha^T} \cdot \overline{\Omega}_i^{k,\alpha} + \sum_{\beta=0}^{s} a^{\alpha\beta} \overline{S}_i^{k,\beta^T} \cdot \overline{\mu}_i^{k,\beta} + Q_i^{k,\alpha} \quad \forall \alpha = 1, \cdots, s-1, \tag{D.3b}$$

$$p_i^{k+1} = \overline{S}_i^{k,s^T} \cdot \overline{\Omega}_i^{k,s} + \sum_{\beta=0}^{s} a^{s\beta} \overline{S}_i^{k,\beta^T} \cdot \overline{\mu}_i^{k,\beta} + Q_i^{k,s} \tag{D.3c}$$

in which $r_i^{k,\alpha}$ is the residue of the DEL equations Eqs. (5a) *and* (5b)*, $a^{\alpha\beta} = w^\beta b^{\beta\alpha}$, $\overline{\Omega}_i^{k,\alpha} = w^\alpha \Delta t \cdot \mathrm{ad}^T_{\overline{v}_i^{k,\alpha}} \cdot \overline{\mu}_i^{k,\alpha} + \overline{\Gamma}_i^{k,\alpha}$, and $Q_i^{k,\alpha} = \omega^\alpha Q_i(t^{k,\alpha})\Delta t$ is the discrete joint force applied to joint i.*

Proof. The Lagrangian of a mechanical system is defined to be

$$\mathcal{L}(q,\dot{q}) = K(q,\dot{q}) - V(q) \tag{D.4}$$

in which $K(q,\dot{q})$ is the kinetic energy and $V(q)$ is the potential energy. It is by the definition of $\overline{F}_i(t)$ and $Q_i(t)$ that

$$\int_0^T \mathcal{F}(t)\cdot \delta q dt - \delta \int_0^T V(q) dt = \int_0^T \sum_{i=1}^n \overline{F}_i(t)\cdot \overline{\eta}_i dt + \int_0^T \sum_{i=1}^n Q_i(t)\cdot \delta q_i dt$$

in which $\overline{\eta}_i = (\delta g_i g_i^{-1})^\vee$. Therefore, the Lagrange-d'Alembert principle Eq. (1) is equivalent to

$$\delta \mathfrak{S} = \delta \int_0^T K(q,\dot{q}) dt + \int_0^T \sum_{i=1}^n \overline{F}_i(t)\cdot \overline{\eta}_i dt + \int_0^T \sum_{i=1}^n Q_i(t)\cdot \delta q_i dt = 0. \tag{D.5}$$

As a result of Eqs. (3) and (D.5), we have

$$\sum_{k=0}^{N-1}\sum_{\alpha=0}^{s} w^\alpha \sum_{i=1}^{n}\Big[\Big\langle \tfrac{\partial K}{\partial q_i}(q^{k,\alpha},\dot{q}^{k,\alpha}), \delta q_i^{k,\alpha}\Big\rangle + \Big\langle \tfrac{\partial K}{\partial \dot{q}_i}(q^{k,\alpha},\dot{q}^{k,\alpha}), \delta \dot{q}_i^{k,\alpha}\Big\rangle + \langle \overline{F}_i(t^{k,\alpha}), \overline{\eta}_i^{k,\alpha}\rangle + \langle Q_i(t^{k,\alpha}), \delta q_i^{k,\alpha}\rangle\Big]\Delta t = 0. \quad \text{(D.6)}$$

Note that the kinetic energy $K(q^{k,\alpha},\dot{q}^{k,\alpha})$ is

$$K(q^{k,\alpha},\dot{q}^{k,\alpha}) = \frac{1}{2}\sum_{j=1}^{n} \overline{v}_j^{k,\alpha\,T}\overline{M}_j^{k,\alpha}\overline{v}_j^{k,\alpha} \quad \text{(D.7)}$$

in which $\overline{M}_i^{k,\alpha} \in \mathbb{R}^{6\times 6}$ is the spatial inertia matrix and $\overline{v}_i^{k,\alpha} \in \mathbb{R}^6$ is the spatial velocity. Using Eqs. (C.10b), (D.1) and (D.7), we obtain

$$\begin{aligned}\frac{\partial K}{\partial \dot{q}_i}(q^{k,\alpha},\dot{q}^{k,\alpha}) =& \sum_{j=1}^{n}\frac{\partial \overline{v}_j^{k,\alpha}}{\partial \dot{q}_i}^T \overline{M}_j^{k,\alpha}\overline{v}_j^{k,\alpha} \\ =& \overline{S}_i^{k,\alpha\,T}\overline{M}_i^{k,\alpha}\overline{v}_i^{k,\alpha} + \sum_{j\in\mathrm{des}(i)} \overline{S}_i^{k,\alpha\,T}\overline{M}_j^{k,\alpha}\overline{v}_j^{k,\alpha} \\ =& \overline{S}_i^{k,\alpha\,T}\overline{\mu}_i^{k,\alpha}.\end{aligned} \quad \text{(D.8)}$$

In a similar way, as a result of Eqs. (C.14b), (C.11b), (C.12), (D.1) and (D.7), a tedious but straightforward algebraic manipulation results in

$$\begin{aligned}\frac{\partial K}{\partial q_i}(q^{k,\alpha},\dot{q}^{k,\alpha}) =& \sum_{j\in\mathrm{des}(i)\cup\{i\}}\Big[\mathrm{ad}_{\overline{S}_i^{k,\alpha}}(\overline{v}_j^{k,\alpha} - \overline{v}_i^{k,\alpha}) - \mathrm{ad}_{\overline{S}_i^{k,\alpha}}\overline{v}_j^{k,\alpha}\Big]^T \overline{M}_j^{k,\alpha} v_j^{k,\alpha} \\ =& S_i^{k,\alpha\,T}\mathrm{ad}_{\overline{v}_i}^T \cdot \overline{\mu}_i^{k,\alpha} \\ =& \dot{\overline{S}}_i^{k,\alpha\,T}\overline{\mu}_i^{k,\alpha}.\end{aligned} \quad \text{(D.9)}$$

In addition, using Eqs. (C.4) and (D.2) and $\overline{F}_i^{k,\alpha} = w^\alpha \overline{F}_i(t^{k,\alpha})\Delta t$, we obtain

$$\begin{aligned}\sum_{i=1}^{n}\langle w^\alpha \overline{F}_i(t^{k,\alpha})\Delta t, \overline{\eta}_i^{k,\alpha}\rangle =& \sum_{i=1}^{n}\langle w^\alpha \overline{F}_i(t^{k,\alpha})\Delta t, \overline{S}_i^{k,\alpha}\cdot\delta q_i^{k,\alpha} + \sum_{j\in\mathrm{anc}(i)}\overline{S}_j^{k,\alpha}\cdot q_j^{k,\alpha}\rangle \\ =& \sum_{i=1}^{n}\langle \overline{F}_i^{k,\alpha} + \sum_{j\in\mathrm{des}(i)}\overline{F}_j^{k,\alpha}, \overline{S}_i^{k,\alpha}\cdot\delta q_i^{k,\alpha}\rangle \\ =& \sum_{i=1}^{n}\langle \overline{\Gamma}_i^{k,\alpha}, \overline{S}_i^{k,\alpha}\cdot\delta q_i^{k,\alpha}\rangle \\ =& \sum_{i=1}^{n}\langle \overline{S}_i^{k,\alpha\,T}\overline{\Gamma}_i^{k,\alpha}, \delta q_j^{k,\alpha}\rangle.\end{aligned} \quad \text{(D.10)}$$

From Eq. (2), we obtain

$$\delta\dot{q}_i^{k,\alpha} = \frac{1}{\Delta t}\sum_{\beta=0}^{s} b^{\alpha\beta}\cdot\delta q_i^{k,\beta}. \tag{D.11}$$

Substituting Eqs. (D.8) to (D.10) into Eq. (D.6) and simplifying the resulting equation with Eq. (D.11) as well as the chain rule, we obtain

$$\sum_{k=0}^{N-1}\sum_{\alpha=0}^{s}\sum_{i=1}^{n}\langle \overline{S}_i^{k,\alpha^T}\cdot\overline{\Omega}_i^{k,\alpha} + \sum_{\beta=0}^{s} a^{\alpha\beta}\overline{S}_i^{k,\beta^T}\cdot\overline{\mu}_i^{k,\beta} + Q_i^{k,\alpha}, \delta q_i^{k,\alpha}\rangle = 0$$

in which $a^{\alpha\beta} = w^\beta b^{\beta\alpha}$, $\overline{\Omega}_i^{k,\alpha} = w^\alpha\Delta t\cdot\mathrm{ad}^T_{\overline{v}_i^{k,\alpha}}\cdot\overline{\mu}_i^{k,\alpha} + \overline{\Gamma}_i^{k,\alpha}$ and $Q_i^{k,\alpha} = \omega^\alpha Q_i(t^{k,\alpha})\Delta t$. The equation above is equivalent to requiring

$$p_i^k + \overline{S}_i^{k,0^T}\cdot\overline{\Omega}_i^{k,0} + \sum_{\beta=0}^{s} a^{0\beta}\overline{S}_i^{k,\beta^T}\cdot\overline{\mu}_i^{k,\beta} + Q_i^{k,0} = 0,$$

$$\overline{S}_i^{k,\alpha^T}\cdot\overline{\Omega}_i^{k,\alpha} + \sum_{\beta=0}^{s} a^{\alpha\beta}\overline{S}_i^{k,\beta^T}\cdot\overline{\mu}_i^{k,\beta} + Q_i^{k,\alpha} = 0 \quad \forall\alpha = 1,\cdots,s-1,$$

$$p_i^{k+1} = \overline{S}_i^{k,s^T}\cdot\overline{\Omega}_i^{k,s} + \sum_{\beta=0}^{s} a^{s\beta}\overline{S}_i^{k,\beta^T}\cdot\overline{\mu}_i^{k,\beta} + Q_i^{k,s}.$$

This completes the proof. □

D.2 Proof of Proposition 2

In Section 3.2, we make the assumption on the discrete impulse $\overline{F}_i^{k,\alpha}$ and discrete joint force $Q_i^{k,\alpha}$ as follows.

Assumption 1. *Let $u(t)$ be control inputs of the mechanical system, we assume that the discrete impulse $\overline{F}_i^{k,\alpha}$ and discrete joint force $Q_i^{k,\alpha}$ can be respectively formulated as $\overline{F}_i^{k,\alpha} = \overline{F}_i^{k,\alpha}(g_i^{k,\alpha}, \overline{v}_i^{k,\alpha}, u^{k,\alpha})$ and $Q_i^{k,\alpha} = Q_i^{k,\alpha}(q_i^{k,\alpha}, \dot{q}_i^{k,\alpha}, u^{k,\alpha})$ in which $u^{k,\alpha} = u(t^{k,\alpha})$.*

From the notion of the spatial variation in Section C.2, we have the following proposition for the Newton direction computation, which is later used in the proof of Proposition 2.

Proposition D.1. *If $\delta q_i^{k,\alpha}$ is the Newton direction for $q_i^{k,\alpha}$, $r_i^{k,\alpha}$ is the residue of the DEL equations Eqs.* (7a) *and* (7b)*, and Assumption 1 holds, the computation of the Newton direction $\delta q_i^{k,\alpha}$ is equivalent to requiring*

$$\overline{\delta}\overline{\mu}_i^{k,\alpha} = \overline{M}_i^{k,\alpha}\overline{\delta}\overline{v}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)}\left(\overline{\delta}\overline{\mu}_j^{k,\alpha} - \mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}}\overline{S}_j^{k,\alpha}\cdot\delta q_j^{k,\alpha}\right)$$
$$\forall\alpha = 0,\ 1,\ \cdots,\ s, \tag{D.12a}$$

$$\overline{\delta}\,\overline{\Gamma}_i^{k,\alpha} = \big(\mathbb{D}_1\overline{F}_i^{k,\alpha} + \mathrm{ad}^D_{\overline{F}_i^{k,\alpha}} - \mathrm{ad}_{\overline{v}_i^{k,\alpha}}\big)\cdot\overline{\eta}_i^{k,\alpha} + \mathbb{D}_2\overline{F}_i^{k,\alpha}\cdot\overline{\delta}\overline{v}_i^{k,\alpha} + \sum_{j\in\mathrm{chd}(i)}\big(\overline{\delta}\,\overline{\Gamma}_j^{k,\alpha} - \mathrm{ad}^D_{\overline{\Gamma}_j^{k,\alpha}}\overline{S}_j^{k,\alpha}\cdot\delta q_j^{k,\alpha}\big)\quad\forall\alpha=0,\,1,\,\cdots,\,s-1,\tag{D.12b}$$

$$\overline{\delta}\,\overline{\Omega}_i^{k,\alpha} = \omega^\alpha\Delta t\cdot\big(\mathrm{ad}^T_{\overline{v}_i^{k,\alpha}}\cdot\overline{\delta}\overline{\mu}_i^{k,\alpha} + \mathrm{ad}^D_{\overline{\mu}_i^{k,\alpha}}\overline{\delta}\overline{v}_i^{k,\alpha}\big) + \overline{\delta}\,\overline{\Gamma}_i^{k,\alpha}\quad\forall\alpha=0,\,1,\,\cdots,\,s-1,\tag{D.12c}$$

$$\overline{S}_i^{k,\alpha\,T}\overline{\delta}\,\overline{\Omega}_i^{k,\alpha} + \sum_{\beta=0}^{s} a^{\alpha\beta}\overline{S}_i^{k,\beta\,T}\overline{\delta}\overline{\mu}_i^{k,\beta} + \mathbb{D}_1 Q_i^{k,\alpha}\cdot\delta q_i^{k,\alpha} + \mathbb{D}_2 Q_i^{k,\alpha}\cdot\delta\dot{q}_i^{k,\alpha} = -r_i^{k,\alpha}\quad\forall\alpha=0,\,1,\,\cdots,\,s-1.\tag{D.12d}$$

in which $\overline{\delta}\overline{v}_i^{k,\alpha}$, $\overline{\delta}\overline{\mu}_i^{k,\alpha}$, $\overline{\delta}\,\overline{\Gamma}_i^{k,\alpha}$ and $\overline{\delta}\,\overline{\Omega}_i^{k,\alpha}$ are the spatial variations of $\overline{v}_i^{k,\alpha}$, $\overline{\mu}_i^{k,\alpha}$, $\overline{\Gamma}_i^{k,\alpha}$ and $\overline{\Omega}_i^{k,\alpha}$, respectively. Note that $\delta q_i^{k,0}=0$ and $\overline{\eta}_i^{k,0}=0$ though $\overline{\delta}\overline{v}_i^{k,0}\neq 0$.

Proof. Eqs. (D.12a) and (D.12c) are respectively the same as Eqs. (C.22) and (C.24), thus we only need to prove Eqs. (D.12b) and (D.12d).

From Assumption 1, we have $\overline{F}_i^{k,\alpha} = \overline{F}_i^{k,\alpha}(g_i^{k,\alpha}, \overline{v}_i^{k,\alpha}, u^{k,\alpha})$, and since $\delta u_i^{k,\alpha}=0$, we obtain $\delta\overline{F}_i^{k,\alpha}$ as

$$\delta\overline{F}_i^{k,\alpha} = \mathbb{D}_1\overline{F}_i^{k,\alpha}\cdot\overline{\eta}_i^{k,\alpha} + \mathbb{D}_2\overline{F}_i^{k,\alpha}\cdot\delta\overline{v}_i^{k,\alpha}.$$

According to Eq. (C.16), the spatial variation $\overline{\delta}\,\overline{F}_i^{k,\alpha}$ is

$$\overline{\delta}\,\overline{F}_i^{k,\alpha} = \mathbb{D}_1\overline{F}_i^{k,\alpha}\cdot\overline{\eta}_i^{k,\alpha} + \mathbb{D}_2\overline{F}_i^{k,\alpha}\cdot\delta\overline{v}_i^{k,\alpha} + \mathrm{ad}^T_{\overline{\eta}_i^{k,\alpha}}\overline{F}_i^{k,\alpha}.$$

Since $\delta\overline{v}_i^{k,\alpha} = \overline{\delta}\overline{v}_i^{k,\alpha} + \mathrm{ad}_{\overline{\eta}_i^{k,\alpha}}\overline{v}_i^{k,\alpha}$, $\mathrm{ad}_{\overline{v}_i^{k,\alpha}}\overline{\eta}_i^{k,\alpha} = -\mathrm{ad}_{\overline{\eta}_i^{k,\alpha}}\overline{v}_i^{k,\alpha}$ as well as $\mathrm{ad}^T_{\overline{\eta}_i^{k,\alpha}}\overline{F}_i^{k,\alpha} = \mathrm{ad}^D_{\overline{F}_i^{k,\alpha}}\overline{\eta}_i^{k,\alpha}$, the equation above is equivalent to

$$\overline{\delta}\,\overline{F}_i^{k,\alpha} = \big(\mathbb{D}_1\overline{F}_i^{k,\alpha} + \mathrm{ad}^D_{\overline{F}_i^{k,\alpha}} - \mathbb{D}_2\overline{F}_i^{k,\alpha}\mathrm{ad}_{\overline{v}_i^{k,\alpha}}\big)\cdot\overline{\eta}_i^{k,\alpha} + \mathbb{D}_2\overline{F}_i^{k,\alpha}\cdot\overline{\delta}\overline{v}_i^{k,\alpha}.$$

Substitute the equation above into Eq. (C.23), the result of which is Eq. (D.12b).

As for the proof of Eq. (D.12d), from Eqs. (7a) and (7b), the Newton direction $\delta q_i^{k,\alpha}$ requires that

$$\delta\big(S_i^{k,\alpha\,T}\overline{\Omega}_i\big) + \sum_{\beta=0}^{s} a^{\alpha\beta}\delta\big(S_i^{k,\beta\,T}\overline{\mu}_i^{k,\beta}\big) + \mathbb{D}_1 Q_i^{k,\alpha}\cdot\delta q_i^{k,\alpha} + \mathbb{D}_2 Q_i^{k,\alpha}\cdot\delta\dot{q}_i^{k,\alpha} = -r_i^{k,\alpha}\quad\forall\alpha=0,\,1,\,\cdots,\,s-1.\tag{D.13}$$

As a result of Eqs. (C.17) and (C.18), we have $\delta\big(\overline{S}_i^{k,\alpha\,T}\overline{\mu}_i^{k,\alpha}\big) = \overline{S}_i^{k,\alpha\,T}\overline{\delta}\overline{\mu}_i^{k,\alpha}$ and $\delta\big(\overline{S}_i^{k,\alpha\,T}\overline{\Omega}_i^{k,\alpha}\big) = \overline{S}_i^{k,\alpha\,T}\overline{\delta}\,\overline{\Omega}_i^{k,\alpha}$, with which and Eq. (D.13), we obtain Eq. (D.12d). This completes the proof. □

In Section 3.2, Proposition 2 to compute the Newton direction is stated as follows, for which note that the higher-order variational integrator has $s+1$ control points and the mechanical system has n degrees of freedom.

Proposition 2. *For higher-order variational integrators of unconstrained mechanical systems, if Assumption 1 holds and* ${\mathcal{J}^k}^{-1}(\overline{q}^k)$ *exists, the Newton direction* $\delta\overline{q}^k = -{\mathcal{J}^k}^{-1}(\overline{q}^k)\cdot r^k$ *can be computed with Algorithm B.1 in* $O(s^3 n)$ *time.*

Proof. The proof consists of proving the correctness and the $O(n)$ complexity of the algorithms.

For each $j \in \mathrm{chd}(i)$, we suppose that there exists $D_j^{k,\alpha\rho}$, $G_j^{k,\alpha\nu}$, $l_j^{k,\alpha}$ and $\Pi_j^{k,\alpha\rho}$, $\Psi_j^{k,\alpha\nu}$, $\zeta_j^{k,\alpha}$ such that

$$\overline{\delta}\overline{\mu}_j^{k,\alpha} = \sum_{\rho=0}^{s} D_j^{k,\alpha\rho}\cdot\overline{\delta}\overline{v}_j^{k,\rho} + \sum_{\nu=1}^{s} G_j^{k,\alpha\nu}\cdot\overline{\eta}_j^{k,\nu} + l_j^{k,\alpha} \qquad \forall\alpha = 0,\,1,\,\cdots,\,s, \tag{D.14}$$

$$\overline{\delta}\overline{\Gamma}_j^{k,\alpha} = \sum_{\rho=0}^{s} \Pi_j^{k,\alpha\rho}\cdot\overline{\delta}\overline{v}_j^{k,\rho} + \sum_{\nu=1}^{s} \Psi_j^{k,\alpha\nu}\cdot\overline{\eta}_j^{k,\nu} + \zeta_j^{k,\alpha} \qquad \forall\alpha = 0,\,1,\,\cdots,\,s-1. \tag{D.15}$$

According to Eqs. (C.3), (C.19) and (D.11), $\overline{\delta}\overline{v}_j^{k,\rho}$ and $\overline{\eta}_j^{k,\nu}$ can be respectively computed as

$$\overline{\eta}_j^{k,\nu} = \overline{\eta}_i^{k,\nu} + \overline{S}_j^{k,\nu}\cdot\delta q_j^{k,\nu} \tag{D.16}$$

and

$$\overline{\delta}\overline{v}_j^{k,\rho} = \overline{\delta}\overline{v}_i^{k,\rho} + \dot{\overline{S}}_j^{k,\rho}\cdot\delta q_j^{k,\rho} + \frac{1}{\Delta t}\overline{S}_j^{k,\rho}\sum_{\gamma=1}^{s} b^{\rho\gamma}\cdot\delta q_j^{k,\gamma} \tag{D.17}$$

for which note that $\delta q_j^{k,0}=0$. Substitute Eqs. (D.16) and (D.17) into Eq. (D.14), algebraic manipulation shows that

$$\overline{\delta}\overline{\mu}_j^{k,\alpha} = \sum_{\rho=0}^{s} D_j^{k,\alpha\rho}\cdot\overline{\delta}\overline{v}_i^{k,\rho} + \sum_{\nu=1}^{s} G_j^{k,\alpha\nu}\cdot\overline{\eta}_i^{k,\nu} + l_j^{k,\alpha} + \sum_{\gamma=1}^{s} H_j^{k,\alpha\gamma}\delta q_j^{k,\gamma}, \tag{D.18}$$

in which

$$H_j^{k,\alpha\gamma} = D_j^{k,\alpha\gamma}\dot{\overline{S}}_j^{k,\gamma} + G_j^{k,\alpha\gamma}\overline{S}_j^{k,\gamma} + \frac{1}{\Delta t}\sum_{\rho=0}^{s} b^{\rho\gamma} D_j^{k,\alpha\rho}\overline{S}_j^{k,\rho}.$$

In a similar way, using Eqs. (D.15) to (D.17), we also have

$$\overline{\delta}\overline{\Gamma}_j^{k,\alpha} = \sum_{\rho=0}^{s} \Pi_j^{k,\alpha\rho}\cdot\overline{\delta}\overline{v}_i^{k,\rho} + \sum_{\nu=1}^{s} \Psi_j^{k,\alpha\nu}\cdot\overline{\eta}_i^{k,\nu} + \zeta^{k,\alpha} + \sum_{\gamma=1}^{s} \Phi_j^{k,\alpha\gamma}\delta q_j^{k,\gamma} \tag{D.19}$$

in which

$$\Phi_j^{k,\alpha\gamma} = \Pi_j^{k,\alpha\gamma}\dot{\overline{S}}_j^{k,\gamma} + \Psi_j^{k,\alpha\gamma}\overline{S}_j^{k,\gamma} + \frac{1}{\Delta t}\sum_{\rho=0}^{s} b^{\rho\gamma}\Pi_j^{k,\alpha\rho}\overline{S}_j^{k,\rho}.$$

From Eqs. (C.12), (C.24) and (D.17) to (D.19) and

$$\overline{S}_j^{k,\alpha^T}\mathrm{ad}^T_{\overline{S}_j^{k,\alpha}}\overline{\mu}_j^{k,\alpha} = \overline{S}_j^{k,\alpha^T}\mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}}\overline{S}_j^{k,\alpha} = 0,$$

we obtain

$$\overline{S}_j^{k,\alpha^T}\bar{\delta}\,\overline{\Omega}_j^{k,\alpha} = \sum_{\rho=0}^{s}\Theta_j^{k,\alpha\rho}\cdot\bar{\delta}\overline{v}_i^{k,\rho} + \sum_{\nu=1}^{s}\Xi^{k,\alpha\nu}\cdot\overline{\eta}_i^{k,\nu} + \xi_j^{k,\alpha} \tag{D.20}$$

in which

$$\begin{aligned}
\Theta_j^{k,\alpha\rho} &= w^\alpha\Delta t\cdot\big(\dot{\overline{S}}_j^{k,\alpha^T}D_j^{k,\alpha\rho} + \sigma^{\alpha\rho}\overline{S}_j^{k,\alpha^T}\mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}}\big) + \overline{S}_j^{k,\alpha^T}\Pi_j^{k,\alpha\rho},\\
\Xi_j^{k,\alpha\nu} &= w^\alpha\Delta t\cdot\dot{\overline{S}}_j^{k,\alpha^T}G_j^{k,\alpha\nu} + \overline{S}_j^{k,\alpha^T}\Psi_j^{k,\alpha\nu},\\
\xi_j^{k,\alpha} &= w^\alpha\Delta t\cdot\dot{\overline{S}}_j^{k,\alpha^T}l_j^{k,\alpha} + \overline{S}_j^{k,\alpha^T}\zeta_j^{k,\alpha} + \sum_{\gamma=1}^{s}\Big[w^\alpha\Delta t\cdot\big(\dot{\overline{S}}_j{}^{k,\alpha^T}H_j^{k,\alpha\gamma} +\\
&\qquad \sigma^{\alpha\gamma}\overline{S}_j^{k,\alpha^T}\mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}}\dot{\overline{S}}_j^{k,\alpha}\big) + \overline{S}_j^{k,\alpha^T}\Phi_j^{k,\alpha\gamma}\Big]\delta q_j^{k,\gamma},
\end{aligned}$$

and note that $\sigma^{\alpha\rho}$ is given in Eq. (B.2) of Algorithm B.2. Substituting Eqs. (D.11), (D.18) and (D.20) into Eq. (D.12d), we obtain

$$\sum_{\rho=0}^{s}\overline{\Theta}_j^{k,\alpha\rho}\cdot\bar{\delta}\overline{v}_i^{k,\rho} + \sum_{\nu=1}^{s}\overline{\Xi}_j^{k,\alpha\nu}\cdot\overline{\eta}_i^{k,\nu} + \bar{\xi}_j^{k,\alpha} + \sum_{\gamma=1}^{s}\Lambda_j^{k,\alpha\gamma}\cdot\delta q_j^{k,\gamma} = -r_j^{k,\alpha}$$
$$\forall\alpha = 0,\,1,\,\cdots,\,s-1. \tag{D.21}$$

in which

$$\begin{aligned}
\overline{\Theta}_j^{k,\alpha\rho} &= \Theta_j^{k,\alpha\rho} + \sum_{\beta=0}^{s}a^{\alpha\beta}\overline{S}_j^{k,\beta^T}D_j^{k,\beta\rho},\\
\overline{\Xi}_j^{k,\alpha\nu} &= \Xi_j^{k,\alpha\nu} + \sum_{\beta=0}^{s}a^{\alpha\beta}\overline{S}_j^{k,\beta^T}G_j^{k,\beta\nu},\\
\bar{\xi}_j^{k,\alpha} &= w^\alpha\Delta t\cdot\dot{\overline{S}}_j^{k,\alpha^T}l_j^{k,\alpha} + \overline{S}_j^{k,\alpha^T}\zeta_j^{k,\alpha} + \sum_{\beta=0}^{s}a^{\alpha\beta}\overline{S}_j^{k,\beta^T}l_j^{k,\beta},\\
\Lambda_j^{k,\alpha\gamma} &= w^\alpha\Delta t\cdot\dot{\overline{S}}_j^{k,\alpha^T}H_j^{k,\alpha\gamma} + \overline{S}_j^{k,\alpha^T}\Phi_j^{k,\alpha\gamma} + \sum_{\beta=0}^{s}a^{\alpha\beta}\overline{S}_j^{k,\beta^T}H_j^{k,\beta\gamma} +\\
&\qquad \sigma^{\alpha\gamma}\left(\mathbb{D}_1Q_j^{k,\alpha} + w^\alpha\Delta t\cdot\overline{S}_j^{k,\alpha^T}\mathrm{ad}^D_{\overline{\mu}_j^{k,\alpha}}\dot{\overline{S}}_j^{k,\alpha}\right) + \frac{1}{\Delta t}b^{\alpha\gamma}\cdot\mathbb{D}_2Q_j^{k,\alpha}.
\end{aligned}$$

For notational convenience, we define $\varDelta_j^{k,\alpha}$ to be

$$\varDelta_j^{k,\alpha} = \sum_{\rho=0}^{s} \overline{\varTheta}_j^{k,\alpha\rho} \cdot \bar{\delta}\overline{v}_i^{k,\rho} + \sum_{\nu=1}^{s} \overline{\varXi}_j^{k,\alpha\nu} \cdot \overline{\eta}_i^{k,\nu} + \overline{\xi}_j^{k,\alpha} \qquad \forall \alpha = 0,\, 1,\, \cdots,\, s-1. \quad \text{(D.22)}$$

such that Eq. (D.21) is rewritten as

$$\sum_{\gamma=1}^{s} \varLambda_j^{k,\alpha\gamma} \cdot \delta q_j^{k,\gamma} = -r_j^{k,\alpha} - \varDelta_j^{k,\alpha} \qquad \forall \alpha = 0,\, 1,\, \cdots,\, s-1. \qquad \text{(D.23)}$$

In addition, if we further define $\varLambda_j^k$, r_j^k, $\varDelta_j^k$ and $\delta\overline{q}_j^k$ respectively as

$$\begin{aligned}
\varLambda_j^k &= \left[\varLambda_j^{k,\alpha\gamma}\right] \in \mathbb{R}^{s\times s}, \\
r_j^k &= \left[r_j^{k,0}\ r_j^{k,1}\ \cdots\ r_j^{k,s-1}\right]^T \in \mathbb{R}^s, \\
\varDelta_j^k &= \left[\varDelta_j^{k,0}\ \varDelta_j^{k,1}\ \cdots\ \varDelta_j^{k,s-1}\right]^T \in \mathbb{R}^s, \\
\delta\overline{q}_j^k &= \left[\delta q_j^{k,1}\ \delta q_j^{k,2}\ \cdots\ \delta q_j^{k,s}\right]^T \in \mathbb{R}^s,
\end{aligned}$$

in which $0 \le \alpha \le s-1$ and $1 \le \gamma \le s$, then Eq. (D.23) is equivalent to requiring

$$\varLambda_j^k \cdot \delta\overline{q}_j^k = -r_j^k - \varDelta_j^k. \qquad \text{(D.24)}$$

in which $\varLambda_j^k$ is invertible since $\mathcal{J}^{k^{-1}}(\overline{q}^k)$ exists. From Eq. (D.24), we obtain

$$\delta\overline{q}_j^k = -\varLambda_j^{k^{-1}}(r_j^k + \varDelta_j^k).$$

If $\varLambda_j^{k^{-1}}$ is explicitly written as $\varLambda_j^{k^{-1}} = \left[\overline{\varLambda}_j^{k,\gamma\varrho}\right] \in \mathbb{R}^{s\times s}$ in which $1 \le \gamma \le s$ and $0 \le \varrho \le s-1$, expanding the equation above, we obtain

$$\delta q_j^{k,\gamma} = -\sum_{\varrho=0}^{s-1} \overline{\varLambda}_j^{k,\gamma\varrho} \left(r_j^{k,\varrho} + \varDelta_j^{k,\varrho}\right) \qquad \forall \gamma = 1,\, 2,\, \cdots,\, s. \qquad \text{(D.25)}$$

Substitute Eq. (D.22) into Eq. (D.25), the result is

$$\delta q_j^{k,\gamma} = \sum_{\rho=0}^{s} X_j^{k,\gamma\rho} \cdot \bar{\delta}\overline{v}_i^{k,\rho} + \sum_{\nu=1}^{s} Y_j^{k,\gamma\nu} \cdot \overline{\eta}_i^{k,\nu} + y_j^{k,\gamma} \qquad \text{(D.26)}$$

in which

$$X_j^{k,\gamma\rho} = -\sum_{\varrho=0}^{s-1} \overline{\Lambda}_j^{k,\gamma\varrho} \cdot \overline{\Theta}_j^{k,\varrho\rho},$$

$$Y_j^{k,\gamma\nu} = -\sum_{\varrho=0}^{s-1} \overline{\Lambda}_j^{k,\gamma\varrho} \cdot \overline{\Xi}_j^{k,\varrho\nu},$$

$$y_j^{k,\gamma} = -\sum_{\varrho=0}^{s-1} \overline{\Lambda}_j^{k,\gamma\varrho} \left(r_j^{k,\varrho} + \overline{\xi}_j^{k,\varrho} \right).$$

Making use of Eqs. (D.18) and (D.26) and canceling out $\delta q_j^{k,\gamma}$, we obtain

$$\overline{\delta\mu}_j^{k,\alpha} - \mathrm{ad}^D_{\mu_j^{k,\alpha}} \overline{S}_j^{k,\alpha} \cdot \delta q_j^{k,\alpha} = \sum_{\rho=0}^{s} \overline{D}_j^{k,\rho} \cdot \overline{\delta v}_i^{k,\rho} + \sum_{\nu=1}^{s} \overline{G}_j^{k,\alpha\nu} \cdot \overline{\eta}_i^{k,\nu} + \overline{l}_j^{k,\alpha} \tag{D.27}$$

in which $\alpha = 0,\ 1,\ \cdots,\ s$, and

$$\overline{D}_j^{k,\rho} = D_j^{k,\rho} + \sum_{\gamma=1}^{s} H_j^{k,\alpha\gamma} X_j^{k,\gamma\rho} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\mu_j^{k,\alpha}} \overline{S}_j^{k,\alpha} X_j^{k,\alpha\rho}, \tag{D.28a}$$

$$\overline{G}_j^{k,\nu} = G_j^{k,\alpha\nu} + \sum_{\gamma=1}^{s} H_j^{k,\alpha\gamma} Y_j^{k,\gamma\nu} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\mu_j^{k,\alpha}} \overline{S}_j^{k,\alpha} Y_j^{k,\alpha\nu}, \tag{D.28b}$$

$$\overline{l}_j^{k,\alpha} = l_j^{k,\alpha} + \sum_{\gamma=1}^{s} H_j^{k,\alpha\gamma} y_j^{k,\gamma} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\mu_j^{k,\alpha}} \overline{S}_j^{k,\alpha} y_j^{k,\alpha}, \tag{D.28c}$$

and note that $\overline{\sigma}^{\alpha 0}$ is given in Eq. (B.2) of Algorithm B.2. In a similar way, using Eqs. (D.19) and (D.26), we obtain

$$\overline{\delta\Gamma}_j^{k,\alpha} - \mathrm{ad}^D_{\Gamma_j^{k,\alpha}} \overline{S}_j^{k,\alpha} \cdot \delta q_j^{k,\alpha} = \sum_{\rho=0}^{s} \overline{\Pi}_j^{k,\alpha\rho} \cdot \overline{\delta v}_j^{k,\rho} + \sum_{\nu=1}^{s} \overline{\Psi}_j^{k,\alpha\nu} \cdot \overline{\eta}_j^{k,\nu} + \overline{\zeta}_j^{k,\alpha} \tag{D.29}$$

in which $\alpha = 1,\ 2,\ \cdots,\ s$, and

$$\overline{\Pi}_j^{k,\alpha\rho} = \Pi_j^{k,\alpha\rho} + \sum_{\gamma=1}^{s} \Phi_j^{k,\alpha\gamma} X_j^{k,\gamma\rho} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\Gamma_j^{k,\alpha}} \overline{S}_j^{k,\alpha} X_j^{k,\alpha\rho}, \tag{D.30a}$$

$$\overline{\Psi}_j^{k,\alpha\nu} = \Psi_j^{k,\alpha\nu} + \sum_{\gamma=1}^{s} \Phi_j^{k,\alpha\gamma} Y_j^{k,\gamma\nu} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\Gamma_j^{k,\alpha}} \overline{S}_j^{k,\alpha} Y_j^{k,\alpha\nu}, \tag{D.30b}$$

$$\overline{\zeta}_j^{k,\alpha} = \zeta_j^{k,\alpha} + \sum_{\gamma=1}^{s} \Phi_j^{k,\alpha\gamma} y_j^{k,\gamma} - \overline{\sigma}^{\alpha 0} \mathrm{ad}^D_{\Gamma_j^{k,\alpha}} \overline{S}_j^{k,\alpha} y_j^{k,\alpha}. \tag{D.30c}$$

Finally, for each $j \in \mathrm{chd}(i)$, substituting Eqs. (D.27) and (D.29) respectively into Eqs. (D.12a) and (D.12b) and applying Eqs. (D.28) and (D.30) to expand $\overline{D}_j^{k,\rho}$, $\overline{G}_j^{k,\nu}$,

$\bar{l}_j^{k,\alpha}$ and $\overline{\Pi}_j^{k,\alpha\rho}$, $\overline{\Psi}_j^{k,\alpha\nu}$, $\bar{\zeta}_j^{k,\alpha}$, we respectively obtain $D_i^{k,\rho}$, $G_i^{k,\nu}$, $l_i^{k,\alpha}$ and $\Pi_i^{k,\alpha\rho}$, $\Psi_i^{k,\alpha\nu}$, $\zeta_i^{k,\alpha}$ as Eqs. (B.1) and (B.3) of Algorithm B.2 such that

$$\bar{\delta}\overline{\mu}_i^{k,\alpha} = \sum_{\rho=0}^{s} D_i^{k,\alpha\rho} \cdot \bar{\delta}\overline{v}_i^{k,\rho} + \sum_{\nu=1}^{s} G_i^{k,\alpha\nu} \cdot \overline{\eta}_i^{k,\nu} + l_i^{k,\alpha} \qquad \forall\alpha = 0,\, 1,\, \cdots,\, s, \tag{D.31}$$

$$\bar{\delta}\overline{\Gamma}_i^{k,\alpha} = \sum_{\rho=0}^{s} \Pi_i^{k,\alpha\rho} \cdot \bar{\delta}\overline{v}_i^{k,\rho} + \sum_{\nu=1}^{s} \Psi_i^{k,\alpha\nu} \cdot \overline{\eta}_i^{k,\nu} + \zeta_i^{k,\alpha} \qquad \forall\alpha = 0,\, 1,\, \cdots,\, s-1. \tag{D.32}$$

In particular, note that even if rigid body i is the leaf node of the tree representation whose $\mathrm{chd}(i) = \emptyset$, there still exists $D_i^{k,\rho}$, $G_i^{k,\nu}$, $l_i^{k,\alpha}$ and $\Pi_i^{k,\alpha\rho}$, $\Psi_i^{k,\alpha\nu}$, $\zeta_i^{k,\alpha}$ from Eqs. (B.1) and (B.3) of Algorithm B.2. Moreover, as long as $D_i^{k,\rho}$, $G_i^{k,\nu}$, $l_i^{k,\alpha}$ and $\Pi_i^{k,\alpha\rho}$, $\Psi_i^{k,\alpha\nu}$, $\zeta_i^{k,\alpha}$ are given for each rigid body i, we can further obtain $X_i^{k,\alpha\rho}$, $Y_i^{k,\alpha\nu}$, $y_i^{k,\alpha}$ following lines 3 to 9 of Algorithm B.2.

In summary, for each rigid body i, we have shown that $X_i^{k,\alpha\rho}$, $Y_i^{k,\alpha\nu}$, $y_i^{k,\alpha}$ as well as $D_i^{k,\rho}$, $G_i^{k,\nu}$, $l_i^{k,\alpha}$ and $\Pi_i^{k,\alpha\rho}$, $\Psi_i^{k,\alpha\nu}$, $\zeta_i^{k,\alpha}$ are computable through the backward pass by Algorithm B.2, and $\delta q_i^{k,\alpha}$ as well as $\overline{\eta}_i^{k,\alpha}$ and $\bar{\delta}\overline{v}_i^{k,\alpha}$ are computable through the forward pass by lines 4 to 15 of Algorithm B.1, which proves the correctness of the algorithms.

In regard to the complexity, Algorithm B.2 has $O(s^2) + O(s^3)$ complexity since there are $O(s^2)$ quantities and the computation of $\Lambda_i^{k,\alpha^{-1}}$ takes $O(s^3)$ time, and thus the backward pass by lines 1 to 3 of Algorithm B.1 totally takes $O(s^3 n + s^2 n)$ time. Moreover, in lines 4 to 15 of Algorithm B.1, the forward pass takes $O(s^2 n)$ time. As a result, the overall complexity of Algorithm B.1 is $O(s^3 n)$, which proves the complexity of the algorithms. □

D.3 Proof of Proposition 3

Proposition 3. *For the kinetic energy $K(q, \dot{q})$ of a mechanical system, $\frac{\partial^2 K}{\partial \dot{q}^2}$, $\frac{\partial^2 K}{\partial \dot{q}\partial q}$, $\frac{\partial^2 K}{\partial q\partial \dot{q}}$, $\frac{\partial^2 K}{\partial q^2}$ can be recursively computed with Algorithm 2 in $O(n^2)$ time.*

Proof. According to Eqs. (D.1), (D.8) and (D.9), we have

$$\frac{\partial K}{\partial \dot{q}_i} = \overline{S}_i^T \Big(\overline{M}_i \overline{v}_i + \sum_{i' \in \mathrm{des}(i)} \overline{M}_{i'} \overline{v}_{i'}\Big) \tag{D.33}$$

and

$$\frac{\partial K}{\partial q_i} = \dot{\overline{S}}_i^T \Big(\overline{M}_i \overline{v}_i + \sum_{i' \in \mathrm{des}(i)} \overline{M}_{i'} \overline{v}_{i'}\Big). \tag{D.34}$$

Since $\overline{M}_i\overline{v}_i$, $\overline{S}_i$ and $\dot{\overline{S}}_i$ only depend on q_j and $\dot{q}_j$ for $j \in \text{anc}(i) \cup \{i\}$, it is straightforward to show from Eqs. (D.33) and (D.34) that the derivatives $\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j}$, $\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j}$, $\frac{\partial^2 K}{\partial q_i \partial \dot{q}_j}$ and $\frac{\partial^2 K}{\partial q_i \partial q_j}$ can be respectively computed as

$$\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j} = \begin{cases} \frac{\partial}{\partial \dot{q}_j}\left(\frac{\partial K}{\partial \dot{q}_i}\right) & j \in \text{anc}(i) \cup \{i\}, \\ \frac{\partial^2 K}{\partial \dot{q}_j \partial \dot{q}_i} & j \in \text{des}(i), \\ 0 & \text{otherwise}, \end{cases} \tag{D.35}$$

$$\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j} = \begin{cases} \frac{\partial}{\partial q_j}\left(\frac{\partial K}{\partial \dot{q}_i}\right) & j \in \text{anc}(i) \cup \{i\}, \\ \frac{\partial^2 K}{\partial q_j \partial \dot{q}_i} & j \in \text{des}(i), \\ 0 & \text{otherwise}, \end{cases} \tag{D.36}$$

$$\frac{\partial^2 K}{\partial q_i \partial \dot{q}_j} = \begin{cases} \frac{\partial}{\partial \dot{q}_j}\left(\frac{\partial K}{\partial q_i}\right) & j \in \text{anc}(i) \cup \{i\}, \\ \frac{\partial^2 K}{\partial \dot{q}_j \partial q_i} & j \in \text{des}(i), \\ 0 & \text{otherwise}, \end{cases} \tag{D.37}$$

$$\frac{\partial^2 K}{\partial q_i \partial q_j} = \begin{cases} \frac{\partial}{\partial q_j}\left(\frac{\partial K}{\partial q_i}\right) & j \in \text{anc}(i) \cup \{i\}, \\ \frac{\partial^2 K}{\partial q_j \partial q_i} & j \in \text{des}(i), \\ 0 & \text{otherwise}. \end{cases} \tag{D.38}$$

Therefore, we only need to consider the derivatives for $j \in \text{anc}(i) \cup \{i\}$, whereas the derivatives for $j \notin \text{anc}(i) \cup \{i\}$ are computed from Eqs. (D.35) to (D.38). In addition, if $j \in \text{anc}(i) \cup \{i\}$, using Eqs. (C.14a), (C.10a), (C.11) and (C.12), we obtain

$$\frac{\partial \overline{M}_i \overline{v}_i}{\partial \dot{q}_j} = \overline{M}_i \overline{S}_j, \tag{D.39}$$

$$\begin{aligned}\frac{\partial \overline{M}_i \overline{v}_i}{\partial q_j} &= -\text{ad}_{\overline{S}_j}^T \overline{M}_i \overline{v}_i - \overline{M}_i \text{ad}_{\overline{S}_j} \overline{v}_i + \overline{M}_i \text{ad}_{\overline{S}_j}(\overline{v}_i - \overline{v}_j) \\ &= \overline{M}_i \dot{\overline{S}}_j - \text{ad}_{\overline{S}_j}^T \overline{M}_i \overline{v}_i \end{aligned} \tag{D.40}$$

$$\frac{\partial \dot{S}_i}{\partial \dot{q}_j} = \text{ad}_{\overline{S}_j} \overline{S}_i, \tag{D.41}$$

$$\frac{\partial \dot{S}_i}{\partial q_j} = \text{ad}_{\overline{v}_i} \text{ad}_{\overline{S}_j} \overline{S}_i + \text{ad}_{\text{ad}_{\overline{S}_j}(\overline{v}_i - \overline{v}_j)} \overline{S}_i. \tag{D.42}$$

For notational clarity, we define $\overline{\mu}_i$, $\overline{\mathcal{M}}_i$, $\overline{\mathcal{M}}_i^A$ and $\overline{\mathcal{M}}_i^B$ as

$$\overline{\mu}_i = \overline{M}_i \overline{v}_i + \sum_{j \in \text{des}(i)} \overline{M}_j \overline{v}_j = \overline{M}_i \overline{v}_i + \sum_{j \in \text{chd}(i)} \overline{\mu}_j, \tag{D.43}$$

$$
\begin{aligned}
\overline{\mathcal{M}}_i &= \overline{M}_i + \sum_{j\in\mathrm{des}(i)} \overline{M}_j = \overline{M}_i + \sum_{j\in\mathrm{chd}(i)} \overline{\mathcal{M}}_j, && \text{(D.44)}\\
\overline{\mathcal{M}}_i^A &= \overline{\mathcal{M}}_i \overline{S}_i, && \text{(D.45)}\\
\overline{\mathcal{M}}_i^B &= \overline{\mathcal{M}}_i \dot{\overline{S}}_i - \mathrm{ad}_{\mu_i}^D \overline{S}_i && \text{(D.46)}
\end{aligned}
$$

which will be used in the derivation of $\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j}$, $\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j}$, $\frac{\partial^2 K}{\partial q_i \partial \dot{q}_j}$ and $\frac{\partial^2 K}{\partial q_i \partial q_j}$.

1) $\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j}$

If $j \in \mathrm{anc}(i) \cup \{i\}$, from Eqs. (D.33), (D.39), (D.44) and (D.45), it is simple to show that

$$
\begin{aligned}
\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j} &= \frac{\partial}{\partial \dot{q}_j}\left(\frac{\partial K}{\partial \dot{q}_i}\right)\\
&= \overline{S}_i^T \Big(\overline{M}_i \overline{S}_j + \sum_{i'\in\mathrm{des}(i)} \overline{M}_{i'} \overline{S}_j\Big)\\
&= \overline{S}_i^T \Big(\overline{M}_i + \sum_{i'\in\mathrm{des}(i)} \overline{M}_{i'}\Big)\overline{S}_j\\
&= \overline{S}_j^T \overline{\mathcal{M}}_i \overline{S}_i\\
&= \overline{S}_j^T \overline{\mathcal{M}}_i^A.
\end{aligned}
\tag{D.47}
$$

2) $\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j}$

If $j \in \mathrm{anc}(i) \cup \{i\}$, using Eqs. (C.7a), (D.33), (D.40), (D.44) and (D.45), we obtain

$$
\begin{aligned}
\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j} &= \frac{\partial}{\partial q_j}\left(\frac{\partial K}{\partial \dot{q}_i}\right)\\
&= \sum_{i'\in\mathrm{des}(i)\cup\{i\}} \Big(\overline{S}_i^T \overline{M}_{i'} \dot{\overline{S}}_j - \overline{S}_i^T \mathrm{ad}_{\overline{S}_j}^T \overline{M}_{i'} \overline{v}_{i'} + \overline{S}_i^T \mathrm{ad}_{\overline{S}_j}^T \overline{M}_{i'} \overline{v}_{i'}\Big)\\
&= \overline{S}_i^T \Big(\overline{M}_i + \sum_{i'\in\mathrm{des}(i)} \overline{M}_{i'}\Big)\dot{\overline{S}}_j\\
&= \dot{\overline{S}}_j^T \overline{\mathcal{M}}_i \overline{S}_i\\
&= \dot{\overline{S}}_j^T \overline{\mathcal{M}}_i^A.
\end{aligned}
\tag{D.48}
$$

3) $\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j}$

If $j \in \mathrm{anc}(i) \cup \{i\}$, using Eqs. (D.34), (D.39), (D.41), (D.43) and (D.44), we obtain

$$\begin{aligned}
\frac{\partial^2 K}{\partial q_i \partial \dot{q}_j} &= \frac{\partial}{\partial \dot{q}_j}\left(\frac{\partial K}{\partial q_i}\right) \\
&= \sum_{i' \in \mathrm{des}(i) \cup \{i\}} \left(\dot{\overline{S}}_i^T \overline{M}_{i'} \overline{S}_j + \overline{S}_i^T \mathrm{ad}_{\overline{S}_j}^T \overline{M}_{i'} \overline{v}_{i'}\right) \\
&= \overline{S}_j^T \Big(\overline{M}_i + \sum_{i' \in \mathrm{des}(i)} \overline{M}_{i'}\Big)\dot{\overline{S}}_i + \Big(\overline{M}_i \overline{v}_i + \sum_{i' \in \mathrm{des}(i)} \overline{M}_{i'} \overline{v}_{i'}\Big)^T \mathrm{ad}_{\overline{S}_j} \overline{S}_i \\
&= \overline{S}_j^T \overline{\mathcal{M}}_i \dot{\overline{S}}_i + \overline{\mu}_i^T \mathrm{ad}_{\overline{S}_j} \overline{S}_i.
\end{aligned}$$

Then simplify the equation above with $\overline{\mu}_i^T \mathrm{ad}_{\overline{S}_j} \overline{S}_i = -\overline{S}_j^T \mathrm{ad}_{\overline{\mu}_i}^D \overline{S}_i$ and Eq. (D.46), the result is

$$\frac{\partial^2 K}{\partial q_i \partial \dot{q}_j} = \overline{S}_j^T \left(\overline{\mathcal{M}}_i \dot{\overline{S}}_i - \mathrm{ad}_{\overline{\mu}_i}^D \overline{S}_i\right) = \overline{S}_j^T \overline{\mathcal{M}}_i^B. \tag{D.49}$$

4) $\frac{\partial^2 K}{\partial q_i \partial q_j}$

If $j \in \mathrm{anc}(i) \cup \{i\}$, using Eqs. (C.12), (D.34), (D.39), (D.40) and (D.42) to (D.44) and $\mathrm{ad}_{\mathrm{ad}_{\overline{v}_i} \overline{S}_j} = \mathrm{ad}_{\overline{v}_i} \mathrm{ad}_{\overline{S}_j} - \mathrm{ad}_{\overline{S}_j} \mathrm{ad}_{\overline{v}_i}$, we obtain

$$\begin{aligned}
\frac{\partial^2 K}{\partial q_i \partial q_j} &= \frac{\partial}{\partial q_j}\left(\frac{\partial K}{\partial q_i}\right) \\
&= \sum_{i' \in \mathrm{des}(i) \cup \{i\}} \Bigg[\left(\overline{M}_{i'} \overline{v}_{i'}\right)^T \Big(\mathrm{ad}_{\overline{v}_i} \mathrm{ad}_{\overline{S}_j} \overline{S}_i - \mathrm{ad}_{\overline{S}_j} \mathrm{ad}_{\overline{v}_i} \overline{S}_i + \\
&\qquad \mathrm{ad}_{\mathrm{ad}_{\overline{S}_j}(\overline{v}_i - \overline{v}_j)} \overline{S}_i\Big) + \dot{\overline{S}}_j^T \overline{M}_{i'} \dot{\overline{S}}_i \Bigg] \\
&= \dot{\overline{S}}_j^T \Big(\overline{M}_i + \sum_{i' \in \mathrm{des}(i)} \overline{M}_{i'}\Big)\dot{\overline{S}}_i + \Big(\overline{M}_i \overline{v}_i + \sum_{i' \in \mathrm{des}(i)} \overline{M}_{i'} \overline{v}_{i'}\Big)^T \mathrm{ad}_{\dot{\overline{S}}_j} \overline{S}_i \\
&= \dot{\overline{S}}_j^T \overline{\mathcal{M}}_i \dot{\overline{S}}_i + \overline{\mu}_i^T \mathrm{ad}_{\dot{\overline{S}}_j} \overline{S}_i.
\end{aligned}$$

Similar to $\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j}$, using $\overline{\mu}_i^T \mathrm{ad}_{\dot{\overline{S}}_j} \overline{S}_i = -\dot{\overline{S}}_j^T \mathrm{ad}_{\overline{\mu}_i}^D \overline{S}_i$ and Eq. (D.46), we obtain

$$\frac{\partial^2 K}{\partial q_i \partial q_j} = \dot{\overline{S}}_j^T \left(\overline{\mathcal{M}}_i \dot{\overline{S}}_i - \mathrm{ad}_{\overline{\mu}_i}^D \overline{S}_i\right) = \dot{\overline{S}}_j^T \overline{\mathcal{M}}_i^B. \tag{D.50}$$

Thus far, we have proved that $\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j}$, $\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j}$, $\frac{\partial^2 K}{\partial q_i \partial \dot{q}_j}$ and $\frac{\partial^2 K}{\partial q_i \partial q_j}$ can be computed using Eqs. (D.35) to (D.38) and (D.47) to (D.50) with which we further have $\frac{\partial^2 K}{\partial \dot{q}^2}$, $\frac{\partial^2 K}{\partial \dot{q} \partial q}$, $\frac{\partial^2 K}{\partial q \partial \dot{q}}$ and $\frac{\partial^2 K}{\partial q^2}$ computed.

As for the complexity of Algorithm 2, it takes $O(n)$ time to pass the tree representation forward to compute g_i, M_i, $\overline{S}_i$, $\overline{v}_i$, $\dot{\overline{S}}_i$ and another $O(n)$ time to pass the tree representation backward to compute $\overline{\mu}_i$, $\overline{\mathcal{M}}_i$, $\overline{\mathcal{M}}_i^A$ and $\overline{\mathcal{M}}_i^B$. In the backward pass, $\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j}$, $\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j}$, $\frac{\partial^2 K}{\partial q_i \partial \dot{q}_j}$ and $\frac{\partial^2 K}{\partial q_i \partial q_j}$ are computed for each i using Eqs. (D.35) to (D.38) and (D.47) to (D.50) which totally takes at most $O(n^2)$ time. Therefore, the complexity of Algorithm 2 is $O(n^2)$. This completes the proof. $\square$

D.4 Proof of Proposition 4

Proposition 4. *If $\mathbf{g} \in \mathbb{R}^3$ is gravity, then for the gravitational potential energy $V_{\mathbf{g}}(q)$, $\frac{\partial^2 V_{\mathbf{g}}}{\partial q^2}$ can be recursively computed with Algorithm 3 in $O(n^2)$ time.*

Proof. It is known that the gravitational potential energy $V_{\mathbf{g}}(q)$ is

$$V_{\mathbf{g}}(q) = -\sum_{i=1}^{n} m_i \cdot \mathbf{g}^T p_i. \tag{D.51}$$

in which $m_i \in \mathbb{R}$ is the mass of rigid body i and $p_i \in \mathbb{R}^3$ is the mass center of rigid body i as well as the origin of frame $\{i\}$. In addition, from Eqs. (C.5a) and (C.5b), we have

$$\frac{\partial p_i}{\partial q_j} = \begin{cases} \hat{\overline{s}}_j p_i + \overline{n}_j & j \in \mathrm{anc}(i) \cup \{i\}, \\ 0 & \text{otherwise}, \end{cases} \tag{D.52a}$$

and

$$\frac{\partial p_j}{\partial q_i} = \begin{cases} \hat{\overline{s}}_i p_j + \overline{n}_i & j \in \mathrm{des}(i) \cup \{i\}, \\ 0 & \text{otherwise}, \end{cases} \tag{D.52b}$$

in which $\overline{s}_i, \overline{n}_i \in \mathbb{R}^3$ and $\overline{S}_i = \begin{bmatrix} \overline{s}_i^T & \overline{n}_i^T \end{bmatrix}^T \in \mathbb{R}^6$ is the spatial Jacobian of joint i. From Eqs. (D.52b) and (D.51), algebraic manipulation gives

$$\frac{\partial V_{\mathbf{g}}}{\partial q_i} = -\overline{S}_i^T \left(m_i \begin{bmatrix} \hat{p}_i \mathbf{g} \\ \mathbf{g} \end{bmatrix} + \sum_{i' \in \mathrm{des}(i)} m_{i'} \begin{bmatrix} \hat{p}_{i'} \mathbf{g} \\ \mathbf{g} \end{bmatrix} \right). \tag{D.53}$$

Moreover, observe that $\overline{S}_i$ and p_i only depends on q_j for $j \in \mathrm{anc}(i) \cup \{i\}$, we obtain from Eq. (D.53) that

$$\frac{\partial^2 V_{\mathbf{g}}}{\partial q_i \partial q_j} = \begin{cases} \frac{\partial}{\partial q_j}\left(\frac{\partial V_{\mathbf{g}}}{\partial q_i}\right) & j \in \mathrm{anc}(i) \cup \{i\}, \\ \frac{\partial^2 V_{\mathbf{g}}}{\partial q_j \partial q_i} & j \in \mathrm{des}(i), \\ 0 & \text{otherwise}, \end{cases} \tag{D.54}$$

which means that only $\frac{\partial^2 V_{\mathbf{g}}}{\partial q_i \partial q_j}$ for $j \in \mathrm{anc}(i) \cup \{i\}$ needs to be explicitly computed. If $j \in \mathrm{anc}(i) \cup \{i\}$, using Eqs. (C.7a), (D.52a) and (D.53) as well as the equality $\hat{a}b = -\hat{b}a$ for any $a, b \in \mathbb{R}^3$, we obtain

$$\begin{aligned} \frac{\partial^2 V_{\mathbf{g}}}{\partial q_i \partial q_j} &= \frac{\partial}{\partial q_j}\left(\frac{\partial V_{\mathbf{g}}}{\partial q_i}\right) \\ &= \sum_{i' \in \mathrm{des}(i) \cup \{i\}} m_{i'} \left[\overline{s}_i^T \left(\hat{\mathbf{g}} \hat{\overline{s}}_j p_{i'} + \hat{\overline{s}}_j \hat{p}_{i'} \mathbf{g} \right) - \overline{n}_i^T \hat{\mathbf{g}} \overline{s}_j \right]. \end{aligned}$$

In addition, since $\hat{p}_{i'}\hat{\mathbf{g}}\overline{s}_j = -\hat{\mathbf{g}}\hat{\overline{s}}_j p_{i'} - \hat{\overline{s}}_j \hat{p_{i'}}\mathbf{g}$ and $\hat{a}^T = -\hat{a}$ for any $a \in \mathbb{R}^3$, the equation above is equivalent to

$$\frac{\partial^2 V_{\mathbf{g}}}{\partial q_i \partial q_j} = \overline{s}_j^T \hat{\mathbf{g}} \Big[\big(m_i + \sum_{i' \in \mathrm{des}(i)} m_{i'}\big)\overline{n}_i - \big(m_i \hat{p}_i + \sum_{i' \in \mathrm{des}(i)} m_{i'}\hat{p}_{i'}\big)\overline{s}_i \Big] \tag{D.55}$$

If we define

$$\overline{\sigma}_{m_i} = m_i + \sum_{j \in \mathrm{des}(i)} m_j = m_i + \sum_{j \in \mathrm{chd}(i)} \overline{\sigma}_{m_j},$$

$$\overline{\sigma}_{p_i} = m_i p_i + \sum_{j \in \mathrm{des}(i)} m_j p_j = m_i p_i + \sum_{j \in \mathrm{chd}(i)} \overline{\sigma}_{p_j},$$

$$\overline{\sigma}_i^A = \hat{\mathbf{g}} \left(\overline{\sigma}_{m_i} \cdot \overline{n}_i - \hat{\overline{\sigma}}_{p_i} \cdot \overline{s}_i \right),$$

then Eq. (D.55) is further simplified to

$$\frac{\partial^2 V_{\mathbf{g}}}{\partial q_i \partial q_j} = \overline{s}_j^T \hat{\mathbf{g}} \left(\overline{\sigma}_{m_i} \overline{n}_i - \hat{\overline{\sigma}}_{p_i} \overline{s}_i \right) = \overline{s}_j^T \overline{\sigma}_i^A. \tag{D.56}$$

As a result, $\frac{\partial^2 V_{\mathbf{g}}}{\partial q^2}$ can be computed from Eqs. (D.54) and (D.56).

The $O(n^2)$ complexity of Algorithm 3 is as follows: the forward pass to compute g_i and $\overline{S}_i$ and the backward pass to compute $\overline{\sigma}_{m_i}$, $\overline{\sigma}_{p_i}$ and $\overline{\sigma}_i^A$ take $O(n)$ time, respectively; and the computation of $\frac{\partial^2 V_{\mathbf{g}}}{\partial q_i \partial q_j} = \frac{\partial^2 V_{\mathbf{g}}}{\partial q_j \partial q_i} = \overline{s}_j^T \overline{\sigma}_i^A$ totally takes $O(n^2)$ time. Therefore, it can be concluded that Algorithm 3 has $O(n^2)$ complexity. This completes the proof. $\square$

References

1. Taosha Fan, Jarvis Schultz, and Todd D Murphey. Efficient computation of variational integrators in robotic simulation and trajectory optimization. In *International Workshop on the Algorithmic Foundations of Robotics (WAFR)*, submitted, 2018.