IEEE Transactions on Robotics and Automation, vol. 11, pp. 571-584, Aug. 1995

Diagonalized Lagrangian Robot Dynamics

Abhinandan Jain and Guillermo Rodriguez
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive, Pasadena, CA 91109

Abstract

A diagonal equation $\dot{\boldsymbol{\nu}} + \boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu}) = \boldsymbol{\epsilon}$ *for robot dynamics is developed by combining recent mass matrix factorization results [1–7] with classical Lagrangian mechanics. Diagonalization implies that at each fixed time instant the equation at each joint is decoupled from all of the other joint equations. The equation involves two important variables: a vector* $\boldsymbol{\nu}$ *of* total joint rotational rates *and a corresponding vector* $\boldsymbol{\epsilon}$ *of* working joint moments. *The nonlinear Coriolis term* $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu})$ *depends on the joint angles* $\boldsymbol{\theta}$ *and the rates* $\boldsymbol{\nu}$*. The total joint rates* $\boldsymbol{\nu}$ *are related to the relative joint-angle rates* $\dot{\boldsymbol{\theta}}$ *by a linear spatial operator* $\boldsymbol{m}^*(\boldsymbol{\theta})$ *mechanized by a base-to-tip spatially recursive algorithm. The total rate* $\boldsymbol{\nu}(k)$ *at a given joint k reflects, in a very unique sense the total rotational velocity about the joint, and includes the combined effects from all the links between joint k and the manipulator base. This differs from the more traditional joint-angle rates* $\dot{\boldsymbol{\theta}}$ *which only reflect the relative, as opposed to total, rotation about the joints. Similarly, the working moments* $\boldsymbol{\epsilon} = \boldsymbol{\ell T}$ *are related to the applied moments* $\boldsymbol{T}$ *by the spatial operator* $\boldsymbol{\ell}(\boldsymbol{\theta}) = \boldsymbol{m}^{-1}(\boldsymbol{\theta})$ *mechanized by a tip-to-base spatially recursive algorithm. The working moment* $\boldsymbol{\epsilon}(k)$ *at a given joint k is that part of the applied moment* $\boldsymbol{T}(k)$ *which does actual mechanical work, while its other part affects only the non-working internal constraint forces. The diagonal equations are obtained by using the recently developed [1] mass matrix factorization* $\boldsymbol{\mathcal{M}}(\boldsymbol{\theta}) = \boldsymbol{m}(\boldsymbol{\theta})\boldsymbol{m}^*(\boldsymbol{\theta})$ *of the system Lagrangian. The diagonalization is achieved in velocity space. This means that only the velocity variables* $\dot{\boldsymbol{\theta}}$ *are replaced with the new variables* $\boldsymbol{\nu}$*, while the original configuration variables* $\boldsymbol{\theta}$ *are retained. The new joint velocity variables* $\boldsymbol{\nu}$ *can be viewed as time-derivatives of Lagrangian quasi-coordinates, similar to those of classical mechanics. The velocity transformations are shown to always exist for tree-like, articulated multibody systems, and they can be readily implemented using the spatially recursive filtering and smoothing methods [1, 4, 7] advanced by the authors in recent years.*

1 Mass Matrix Factors Diagonalize Lagrange's Equations

The main new result in this paper is a diagonalized equations of motion $\dot{\boldsymbol{\nu}} + \boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu}) = \boldsymbol{\epsilon}$, which embodies in a simple, elegant, diagonal equation the complete dynamical behavior of robotic manipulators, while simultaneously exploiting the computational efficiency of the spatially recursive filtering and smoothing algorithms of [4, 7] to conduct necessary velocity coordinate transformations. The diagonal equations of motion result by combining Lagrangian mechanics with the mass matrix factorization

$$\boldsymbol{\mathcal{M}} = [\boldsymbol{I} + \boldsymbol{H}\boldsymbol{\phi}\boldsymbol{K}]\boldsymbol{D}[\boldsymbol{I} + \boldsymbol{H}\boldsymbol{\phi}\boldsymbol{K}]^* \tag{1.1}$$

in which $\boldsymbol{H}$, $\boldsymbol{\phi}$, $\boldsymbol{D}$ and $\boldsymbol{K}$ are spatial operators mechanized recursively by suitably defined [4,7] spatial filtering and smoothing algorithms. Use of this in the system kinetic energy $\boldsymbol{\mathcal{K}}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) = \frac{1}{2}\dot{\boldsymbol{\theta}}^*\boldsymbol{\mathcal{M}}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}$ results in

$$\mathcal{K}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) = \frac{1}{2}\boldsymbol{\nu}^*\boldsymbol{\nu} \tag{1.2}$$

where $\boldsymbol{\nu} = [\boldsymbol{\nu}(1),\cdots,\boldsymbol{\nu}(\mathcal{N})]$ are a new set of variables related to the joint-angle rates $\dot{\boldsymbol{\theta}}$ by

$$\boldsymbol{\nu} = \boldsymbol{D}^{\frac{1}{2}}[\boldsymbol{I}+\boldsymbol{H\phi K}]^*\dot{\boldsymbol{\theta}} \tag{1.3}$$

In these new variables, the kinetic energy is diagonalized in the sense that it is a simple sum of the squares of the total joint rates $\boldsymbol{\nu}(k)$ over all of the $\mathcal{N}$ joints. This is in contrast to the original expression $\boldsymbol{\mathcal{K}}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) = \frac{1}{2}\dot{\boldsymbol{\theta}}^*\boldsymbol{\mathcal{M}}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}$ which involves the mass matrix $\boldsymbol{\mathcal{M}}(\boldsymbol{\theta})$ as a weighting matrix.

The diagonal equations of motion $\dot{\boldsymbol{\nu}}+\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu})=\boldsymbol{\epsilon}$ are obtained by applying classical Lagrangian mechanics methods to the above diagonalized kinetic energy. In addition to being diagonalized, the kinetic energy can also be thought of as being normalized. This means that the coefficient multiplying each of the terms in the kinetic energy expression (1.2) is identically equal to $\frac{1}{2}$. An alternative set of diagonal equations of motion with un-normalized coefficients are also derived.

The Total Joint Rates Are Time Derivatives of Lagrangian Quasi-Coordinates

The new variables $\boldsymbol{\nu}$ have a physical interpretation as time-derivatives of Lagrangian quasi-coordinates, similar to those typically encountered [8,9] in analytical dynamics. These new variables are related to the original joint-angle velocities $\dot{\boldsymbol{\theta}}$ by means of the configuration-dependent linear transformation $\boldsymbol{m}^*= \boldsymbol{D}^{\frac{1}{2}}[\boldsymbol{I}+\boldsymbol{H\phi K}]^*$ in $\boldsymbol{\nu} = \boldsymbol{m}^*\dot{\boldsymbol{\theta}}$. This means that when the new joint velocity variables $\boldsymbol{\nu}$ are integrated with respect to time, they do not directly result in the joint-angle configuration variables $\boldsymbol{\theta}$. In order to determine the joint angles, it is first necessary to compute the joint-angle velocities $\dot{\boldsymbol{\theta}}$. This requires that the linear transformation $\boldsymbol{m}^*$ above be inverted in order to obtain $\dot{\boldsymbol{\theta}} = (\boldsymbol{m}^*)^{-1}\boldsymbol{\nu}$. At first, inversion of the transformation $\boldsymbol{m}^*$ looks difficult. However, recent factorization results [1,6] make it trivial to perform this inversion. The inverse transformation is given explicitly by $(\boldsymbol{m}^*)^{-1} = [\boldsymbol{I}-\boldsymbol{H\psi K}]^*\boldsymbol{D}^{-\frac{1}{2}}$ where $\boldsymbol{H}$, $\boldsymbol{\psi}$, $\boldsymbol{K}$, and $\boldsymbol{D}$ are spatial operators also mechanized by efficient spatially recursive algorithms [4,7].

There is a similarity between the variables $\boldsymbol{\nu}$ and the angular velocity vector ω typically used to describe the rotational velocity of a single rigid body with respect to an inertial coordinate frame. This similarity can be used to gain insight about the physical meaning of the total joint-rate variables $\boldsymbol{\nu}$. The dynamics of a single rigid body are governed by the equation $\boldsymbol{\mathcal{J}}\dot{\omega}+\omega\times\boldsymbol{\mathcal{J}}\omega = \boldsymbol{T}$, in which $\boldsymbol{\mathcal{J}}$ is the rotational inertia tensor, and $\boldsymbol{T}$ is the vector of applied moments. This equation of motion is considerably simpler and elegant than that which would be obtained by using the system configuration variables, which for a rigid body would be typically the three Euler angles $\boldsymbol{\theta} = [\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\boldsymbol{\theta}_3]$ describing the orientation of the body. Although the dynamics equations are simpler, there is a drawback: direct integration of the angular velocity ω does not produce the body orientation. The angular velocity variables are therefore time-derivatives of quasi-coordinates. They are related to the time derivatives $\dot{\boldsymbol{\theta}}$ of the configuration variables $\boldsymbol{\theta}$ by means of a linear, configuration dependent transformation $\boldsymbol{m}^*(\boldsymbol{\theta})$, which for a rigid body is a trigonometric function of the configuration variables. This means that $\omega = \boldsymbol{m}^*(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}$ and $\dot{\boldsymbol{\theta}} = [\boldsymbol{m}^*(\boldsymbol{\theta})]^{-1}\omega$. Thus, use of the angular velocity ω leads to very simple equations of motion but with the price that the corresponding kinematic equation $\dot{\boldsymbol{\theta}} = [\boldsymbol{m}^*(\boldsymbol{\theta})]^{-1}\omega$ requires inversion of the transformation $\boldsymbol{m}^*(\boldsymbol{\theta})$. In the case of a single rigid body, it is possible to do this analytically and with manageable penalty.

In the case of multiple, linked rigid bodies considered in this paper, the factorization results of [4, 7] enable a similar analytical conversion from the time-derivatives of quasi-coordinates $\boldsymbol{\nu}$ to the joint-angle velocities $\dot{\boldsymbol{\theta}}$. Integrating the joint-angle rates $\dot{\boldsymbol{\theta}}$ with respect to time results in the system configuration. One important difference is that, the transformation $\boldsymbol{m}^*$ relating the new rate coordinates $\boldsymbol{\nu}$ to the joint-angle rates $\dot{\boldsymbol{\theta}}$ depends on dynamics quantities such as link masses and inertias, while for a single rigid-body, it involves only kinematical quantities.

<u>The New Forcing Term Reflects the Working Moments</u>

Another key term in the new equations of motion is the forcing "input" $\boldsymbol{\epsilon} = \mathbf{col}[\boldsymbol{\epsilon}(k)]$ appearing on the right side of the equation. This term is related to the applied moments $\boldsymbol{T}$ by means of the configuration dependent relationship

$$\boldsymbol{\epsilon} = \boldsymbol{m}^{-1}(\boldsymbol{\theta})\boldsymbol{T} = \boldsymbol{D}^{-\frac{1}{2}}\,[\boldsymbol{I} - \boldsymbol{H}\boldsymbol{\psi}\boldsymbol{K})\boldsymbol{T}] \tag{1.4}$$

The operators $\boldsymbol{H}$, $\boldsymbol{\psi}$, $\boldsymbol{K}$ and $\boldsymbol{D}$ are mechanized by an inward filtering operation [4]. The inputs $\boldsymbol{\epsilon}$ also have a physical interpretation. The input $\boldsymbol{\epsilon}(k)$ at the k^{th} joint can be thought of as being that part of the applied moment $\boldsymbol{T}(k)$ that does mechanical work at this joint. This is discussed in more detail later in the paper.

<u>The New Coriolis Term is Computed Both in Closed-Form and Recursively</u>

The Coriolis term $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu})$ in the diagonalized equations of motion depends quadratically on the new velocity variables $\boldsymbol{\nu}$. A closed-form expression for this term is derived that explicitly shows this quadratic dependence. This leads to a relatively simple physical interpretation of the Coriolis forces in the diagonalized equations. The Coriolis term can be computed by means of an inward spatial recursion from the tip of the manipulator to its base. This inward recursion is $O(\mathcal{N})$ in that the number of arithmetical operations increases only linearly with the number of degrees of freedom. Furthermore, the detailed steps in the inward recursion are similar to those required to factor and diagonalize the manipulator mass matrix. Consequently, the effects of the Coriolis term can be easily accounted in the recursions that diagonalize the equations of motion, with very little extra computational cost.

<u>Relationship to Globally Diagonalized Equations</u>

There has been recent interest in finding global coordinate transformations that diagonalize the equations of motion [10–12]. The goal has been to find global coordinate transformations in which both the configuration variables $\boldsymbol{\theta}$ and the corresponding velocity variables $\dot{\boldsymbol{\theta}}$ are replaced by a new set of transformed coordinates. When written in the transformed coordinates, the equations of motion are completely decoupled from each other. Conditions for the existence of such a global transformation are well-established in the theory of non–Euclidean geometry. The globally diagonalizing transformation exists when the metric defined by the mass matrix is free of curvature [13] and in which case the mass matrix is equivalent to one with constant coefficients in the new coordinate system. Unfortunately, as pointed out in [10–12], this is rarely the case for most practical multibody systems.

In contrast, the present paper shows that the goal of diagonalizing in velocity space is always achievable for tree–topology systems. The diagonalizing transformations described here are applied on the tangent space [13] of the configuration manifold instead of the configuration coordinates, i.e., the transformations operate on velocities and time derivatives of configuration variables. While the goals are more modest than in the search for global transformations, on the other hand, diagonalization in velocity space is shown to exist for very general classes of joint-connected multibody systems. Furthermore, explicit spatially recursive filtering and smoothing algorithms are derived to compute efficiently the required velocity-space transformations.

Relationship to The Innovations Approach of Linear Filtering Theory

The quasi-coordinates $\boldsymbol{\nu}$ appearing in the diagonalized equations of motion are closely analogous to the innovations process investigated extensively [14–16] in the area of linear filtering and estimation for state space systems. The innovations process [14] is a central ingredient in factoring, diagonalizing, and inverting state-space system covariance matrices by means of Kalman filtering and smoothing algorithms. The innovations process plays a similar role in the dynamics of mechanical systems [1,4,6]. The analogy between estimation theory and robot dynamics has been one of the central themes investigated by the authors [4,7]. This paper provides an additional chapter in this still unfolding story.

2 Globally Diagonalized Dynamics Are Elegant But Rarely Exist

For a manipulator with $\mathcal{N}$ degrees of freedom, the traditional Lagrangian equations of motion are

$$\boldsymbol{\mathcal{M}}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) = \boldsymbol{T}; \qquad \boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) = \dot{\boldsymbol{\mathcal{M}}}\dot{\boldsymbol{\theta}} - \frac{1}{2}\dot{\boldsymbol{\theta}}^{*}\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}} \tag{2.1}$$

where by definition $\dot{\boldsymbol{\theta}}^{*}\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}} = \mathbf{col}[\dot{\boldsymbol{\theta}}^{*}\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}_i}\dot{\boldsymbol{\theta}}]$, and $\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}_i}$ is the derivative of the mass matrix $\boldsymbol{\mathcal{M}}$ with respect to the joint coordinate $\boldsymbol{\theta}(i)$. The global diagonalization approach seeks to replace the configuration coordinates $\boldsymbol{\theta}$ and their time-derivatives $\dot{\boldsymbol{\theta}}$ with a new set of variables $(\boldsymbol{\vartheta},\dot{\boldsymbol{\vartheta}})$ in which the equations of motion are decoupled. This approach is based on the following assumption, which imposes the very stringent condition that the "square-root" factor of the mass matrix, $\boldsymbol{m}(\boldsymbol{\theta})$, must be the gradient of a global coordinate transformation. This assumption is very rarely satisfied in practice [10–12]. Nonetheless, it is of interest to examine the globally diagonalized equations as an introduction to the locally diagonalized equations discussed in the present paper.

Assumption 1 *There exists a global coordinate transformation* $\boldsymbol{\vartheta} = f(\boldsymbol{\theta}) \in \mathbb{R}^{\mathcal{N}}$ *such that*

$$\nabla_{\boldsymbol{\theta}}\boldsymbol{\vartheta} = \nabla_{\boldsymbol{\theta}} f = \boldsymbol{m}^{*}(\boldsymbol{\theta}) \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}} \tag{2.2}$$

and the matrix function $\boldsymbol{m}(\boldsymbol{\theta})$ *is the "square root" of the mass matrix*

$$\boldsymbol{m}(\boldsymbol{\theta})\boldsymbol{m}^{*}(\boldsymbol{\theta}) = \boldsymbol{\mathcal{M}}(\boldsymbol{\theta}) \tag{2.3}$$

for all $\boldsymbol{\theta}$.

The above assumption requires that the mass matrix factor $\boldsymbol{m}(\boldsymbol{\theta})$ be the gradient of some function $f(\boldsymbol{\theta})$. The requirement that f be a global coordinate transformation implies by definition that f and $\boldsymbol{m}$ must be both differentiable and invertible. It follows from (2.2) that the new generalized velocity vector is $\dot{\boldsymbol{\vartheta}} = \boldsymbol{m}^*(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}$. In terms of this velocity vector $\dot{\boldsymbol{\vartheta}}$, the kinetic energy is $\mathcal{K}(\boldsymbol{\vartheta}, \dot{\boldsymbol{\vartheta}}) = \frac{1}{2}\dot{\boldsymbol{\vartheta}}^*\dot{\boldsymbol{\vartheta}}$.

Lemma 2.1 *When Assumption 1 holds, the equations of motion in the new coordinates* $(\boldsymbol{\vartheta}, \dot{\boldsymbol{\vartheta}})$ *are*

$$\ddot{\boldsymbol{\vartheta}} = \boldsymbol{\epsilon} \qquad \text{where} \quad \boldsymbol{\epsilon} = \boldsymbol{\ell}(\boldsymbol{\theta})\boldsymbol{T} \in \mathbb{R}^{\mathcal{N}} \tag{2.4}$$

with $\boldsymbol{\ell}(\boldsymbol{\theta}) \stackrel{\triangle}{=} \boldsymbol{m}^{-1}(\boldsymbol{\theta})$.

Proof: *Since* $\ddot{\boldsymbol{\vartheta}} = \boldsymbol{m}^*\ddot{\boldsymbol{\theta}} + \dot{\boldsymbol{m}}^*\dot{\boldsymbol{\theta}}$, *then* $\ddot{\boldsymbol{\theta}} = \boldsymbol{\ell}^*[\ddot{\boldsymbol{\vartheta}} - \dot{\boldsymbol{m}}^*\dot{\boldsymbol{\theta}}]$. *Use of this in* (2.1) *and pre-multiplication by* $\boldsymbol{\ell}$ *leads to* $\ddot{\boldsymbol{\vartheta}} + \boldsymbol{\mathcal{C}}(\boldsymbol{\vartheta}, \dot{\boldsymbol{\vartheta}}) = \boldsymbol{\epsilon}$, *where* $\boldsymbol{\mathcal{C}}(\boldsymbol{\vartheta}, \dot{\boldsymbol{\vartheta}}) = \boldsymbol{\ell}\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \dot{\boldsymbol{m}}^*\dot{\boldsymbol{\theta}}$. *However,* $\boldsymbol{\mathcal{C}}(\boldsymbol{\vartheta}, \dot{\boldsymbol{\vartheta}}) = 0$, *since* $\dot{\boldsymbol{\theta}}^*\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}} = \nabla_{\boldsymbol{\theta}}\,[\dot{\boldsymbol{\vartheta}}^*\dot{\boldsymbol{\vartheta}}] = 2[\nabla_{\boldsymbol{\theta}}\dot{\boldsymbol{\vartheta}}^*]\,\dot{\boldsymbol{\vartheta}} = 2[d(\nabla_{\boldsymbol{\theta}}\boldsymbol{\vartheta}^*)/dt]\,\dot{\boldsymbol{\vartheta}} = 2\dot{\boldsymbol{m}}\dot{\boldsymbol{\vartheta}}$ *and* $\boldsymbol{\ell}\boldsymbol{\mathcal{M}}\dot{\boldsymbol{\theta}} = \boldsymbol{\ell}\dot{\boldsymbol{m}}\dot{\boldsymbol{\vartheta}} + \dot{\boldsymbol{m}}^*\dot{\boldsymbol{\theta}}$. ∎

The new equations of motion in (2.4) are very simple. The mass matrix is the identity matrix, and there are no Coriolis forces. The component degrees of freedom are completely decoupled and governed by independent second-order linear differential equations. Thus, the coordinate transformation $f(\boldsymbol{\theta})$ provides globally diagonalizing coordinates $(\boldsymbol{\vartheta}, \dot{\boldsymbol{\vartheta}})$ which replace the earlier $(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ coordinates. Since $\boldsymbol{T}$ is the vector of generalized forces corresponding to the generalized velocities vector $\dot{\boldsymbol{\theta}}$, the principle of virtual work implies that $\boldsymbol{\epsilon}$ is the vector of generalized forces corresponding to the generalized velocities $\dot{\boldsymbol{\vartheta}}$. Note that (2.4) can be obtained alternatively by deriving the Lagrangian equations of motion in the $\boldsymbol{\vartheta}$ coordinate system using the diagonalized expression (1.2) for the kinetic energy.

Now that the simplicity resulting from the global coordinate transformation $f(\boldsymbol{\theta})$ is apparent, we examine conditions under which Assumption 1 is satisfied by multibody systems. The answer is based on a well–established result from non–Euclidean geometry. It is known [13] that the mass matrix $\boldsymbol{\mathcal{M}}$ defines a metric tensor on the configuration manifold. Since tensor quantities are invariant under coordinate transformations, a globally diagonalizing transformation exists if and only if the metric tensor is a Euclidean metric tensor, i.e. one with constant coefficients. A manifold with a Euclidean metric is said to be "flat" and the curvature tensor associated with it is identically zero. The precise necessary conditions for the metric tensor associated with $\boldsymbol{\mathcal{M}}$ to be a Euclidean metric are summarized in the following lemma [11–13].

Lemma 2.2 *For Assumption 1 to hold, it is necessary that the curvature tensor* R *of* $\boldsymbol{\mathcal{M}}$ *vanish, that is, each of the* $\mathcal{N}(\mathcal{N}+1)/2$ *Riemannian symbols of the first kind* R_{hijk} *defined below must vanish.*

$$R_{hijk} = \frac{1}{2}\left[\frac{\partial^2\boldsymbol{\mathcal{M}}_{hk}}{\partial\boldsymbol{\theta}_i\partial\boldsymbol{\theta}_j} + \frac{\partial^2\boldsymbol{\mathcal{M}}_{ij}}{\partial\boldsymbol{\theta}_h\partial\boldsymbol{\theta}_k} - \frac{\partial^2\boldsymbol{\mathcal{M}}_{hj}}{\partial\boldsymbol{\theta}_i\partial\boldsymbol{\theta}_k} - \frac{\partial^2\boldsymbol{\mathcal{M}}_{ik}}{\partial\boldsymbol{\theta}_h\partial\boldsymbol{\theta}_j}\right] + \sum_l \left[\left\{{l \atop ij}\right\}[hk, l] - \left\{{l \atop ik}\right\}[hj, l]\right] \tag{2.5}$$

The quantities $[ij, k]$ *and* $\left\{{k \atop ij}\right\}$ *are the Christoffel symbols of the first and second kind respectively [9] and are computed from appropriate combinations of first derivatives of the mass matrix with respect to the joint angles.* ∎

In practice, the conditions in this lemma are very restrictive, and are rarely satisfied by practical multibody systems [11,12]. Even when they are satisfied, the conditions are extremely difficult to verify, as first and second derivatives of the mass matrix must be computed with respect to the configuration variables $\boldsymbol{\theta}$. The next section describes an alternative approach to diagonalizing the equations of motion that is broadly applicable to complex multibody systems.

3 Diagonalization in Velocity Space is Easier

Instead of diagonalizing globally in configuration space, we look at a diagonalizing transformation in the velocity space. This transformation replaces the joint-angle velocities $\dot{\boldsymbol{\theta}}$ with a new set of velocities $\boldsymbol{\nu}$, without replacing the configuration variables $\boldsymbol{\theta}$. The search for this transformation begins with the following assumption regarding the factorization of the mass matrix.

Assumption 2 *There exists a smooth, differentiable and invertible function $\boldsymbol{m}(\boldsymbol{\theta})$, with inverse denoted by $\boldsymbol{\ell}(\boldsymbol{\theta})$, which factors the mass matrix as $\boldsymbol{\mathcal{M}}(\boldsymbol{\theta}) = \boldsymbol{m}(\boldsymbol{\theta})\boldsymbol{m}^*(\boldsymbol{\theta})$ for all configurations. Unlike the previous Assumption 1, the function $\boldsymbol{m}(\boldsymbol{\theta})$ here need not be the gradient of any function.*

The differentiability of $\boldsymbol{m}$ insures that the vector $\boldsymbol{\nu} = \boldsymbol{m}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}}$ is differentiable. Invertibility of $\boldsymbol{m}(\boldsymbol{\theta})$ insures that time derivatives $\dot{\boldsymbol{\theta}}$ of the configuration variables can be recovered from $\boldsymbol{\nu}$. Under these conditions $\boldsymbol{\nu}$ represents a valid choice for a new generalized velocity vector.

Assumption 2 is much weaker than Assumption 1. One consequence of the fact that $\boldsymbol{m}$ is not the gradient of a function is that the transformed velocity vector $\boldsymbol{\nu}$ is not the time derivative of any vector of configuration variables. Its components $\boldsymbol{\nu}(k)$ are referred to [8] as time derivatives of quasi-coordinates. Integration of the vector $\boldsymbol{\nu}$ with respect to time does not typically lead to the system configuration variables. Nonetheless, finding the system configuration from the transformed velocities $\boldsymbol{\nu}$ is a relatively easy problem. This is done by solving the kinematic equation $\dot{\boldsymbol{\theta}} = \boldsymbol{\ell}(\boldsymbol{\theta})\boldsymbol{\nu}$ for $\dot{\boldsymbol{\theta}}$ and integrating it to update the configuration coordinates. These dynamic and kinematic equations are summarized in the following result.

Lemma 3.1 *The equations of motion using the $(\boldsymbol{\theta}, \boldsymbol{\nu})$ coordinates are*

$$\dot{\boldsymbol{\nu}} + \boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu}) = \boldsymbol{\epsilon} \tag{3.1}$$

with the new Coriolis force vector

$$\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu}) = \boldsymbol{\ell}\left(\dot{\boldsymbol{m}}\boldsymbol{\nu} - \frac{1}{2}\dot{\boldsymbol{\theta}}^* \boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}\right) \tag{3.2}$$

where $\boldsymbol{\epsilon} = \boldsymbol{\ell}(\boldsymbol{\theta})\boldsymbol{T}$. The kinematic equation to obtain the joint-angle rates $\dot{\boldsymbol{\theta}}$ is

$$\dot{\boldsymbol{\theta}} = \boldsymbol{\ell}(\boldsymbol{\theta})\boldsymbol{\nu}$$

Proof: *Similar to that of Lemma 2.1. Replace (2.4) by (3.1), where $\boldsymbol{\mathcal{C}}(\boldsymbol{\nu}, \boldsymbol{\theta}) = \boldsymbol{\ell}\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \dot{\boldsymbol{m}}^*\dot{\boldsymbol{\theta}}$. Use $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \dot{\boldsymbol{m}}\boldsymbol{\nu} + \boldsymbol{m}\dot{\boldsymbol{m}}^*\dot{\boldsymbol{\theta}} - \frac{1}{2}\dot{\boldsymbol{\theta}}^* \boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}$.* ∎

These equations of motion are considerably simpler than the original ones in (2.1). They are quite similar to the globally diagonalized equations in (2.4). The mass matrix here is once again constant and equal to the identity matrix. The main difference is that the Coriolis force term is no longer zero. However, it will be shown later that this Coriolis vector is orthogonal to the generalized velocity vector $\boldsymbol{\nu}$. This implies that the Coriolis term does no mechanical work.

The most critical element leading to the above diagonalized equations is the mass matrix factor $\boldsymbol{m}(\boldsymbol{\theta})$. Clearly, a numerical (e.g. Cholesky-like) factorization of the mass matrix at each configuration can be used to obtain a candidate factor $\boldsymbol{m}(\boldsymbol{\theta})$. This however has the disadvantage that the factors may not smoothly depend on the configuration coordinates and thus might not be differentiable. Even more problematically, numerical factorization procedures provide no systematic way to compute the Coriolis force term $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu})$, since the derivatives of $\boldsymbol{m}$ are required for this purpose. Also, it may not be easy to physically interpret the corresponding transformed variables either.

An alternative that overcomes the limitations of the numerical factorization approach, is provided by the results on the operator factorizations of the manipulator mass matrix discussed in references [1,6]. These factorizations are reviewed in the next section. The factors are constructed using the spatially recursive filtering and smoothing methods of [4, 7], and provide closed-form expressions for $\boldsymbol{m}(\boldsymbol{\theta})$ and its inverse $\boldsymbol{\ell}(\boldsymbol{\theta})$. The algorithms required to do this are efficient, as the number of arithmetical operations increases only linearly with the number of degrees of freedom. In addition, we later derive closed form expressions and computational algorithms for the new Coriolis forces vector $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu})$.

The diagonal equations of motion in (3.1) occupy a middle ground between the globally decoupled equations of motion in Lemma 2.1 and the standard equations of motion in (2.1). While they are not quite as simple as the globally diagonalized equations of Lemma 2.1, they always exist for the broad class of tree-configuration systems.

4 Operator Factorization and Inversion of the Mass Matrix

Recent results [1, 6] have established that the mass matrix can be factored and inverted using methods widely used in linear filtering and estimation theory. These results are summarized by the following identities, whose proof can be found in [1, 6].

Identity 4.1

$$\boldsymbol{\mathcal{M}} = \boldsymbol{H\phi M\phi^* H^*} \tag{4.1a}$$

$$\boldsymbol{\mathcal{M}} = [\boldsymbol{I}+\boldsymbol{H\phi K}]\boldsymbol{D}[\boldsymbol{I}+\boldsymbol{H\phi K}]^* \tag{4.1b}$$

$$[\boldsymbol{I}+\boldsymbol{H\phi K}]^{-1} = \boldsymbol{I}-\boldsymbol{H\psi K} \tag{4.1c}$$

$$\boldsymbol{\mathcal{M}}^{-1} = [\boldsymbol{I}-\boldsymbol{H\psi K}]^*\boldsymbol{D}^{-1}[\boldsymbol{I}-\boldsymbol{H\psi K}] \tag{4.1d}$$

The factorization in (4.1a) is referred to here as the Newton-Euler Operator Factorization, because it is known [6] to be equivalent to the traditional [17] recursive Newton-Euler equations of motion for a serial manipulator. The recursive algorithms embedded in this factorization, while quite useful [17] for inverse dynamics computations, are not by themselves very useful for the diagonalized equations developed in this paper. The primary limitation [1] is that the factors $\boldsymbol{H\phi}$ and $\boldsymbol{\phi}^*\boldsymbol{H}^*$ are neither square nor invertible. Nevertheless, (4.1a) is pivotal for the development of the alternative factorization in (4.1b). This alternative has been referred to [6] as the Innovations Factorization, because of its relationship to the innovations approach [14] of linear filtering theory. This factorization is essential to developing the diagonalized equations of motion.

The Innovations Factorization in (4.1b) is a closed–form, symbolic, (lower-triangular)-(diagonal)-(upper-triangular) LDL^* factorization of the mass matrix $\boldsymbol{\mathcal{M}}$. The factorization is model-based [6] in the sense that the manipulator model itself is used to prescribe each of the computations required. Because of this, every computational step has an immediate physical interpretation. This adds substantial physical insight to the factorization. The factors $[\boldsymbol{I}+\boldsymbol{H\phi K}]$ and $\boldsymbol{D}$ are square with the former being lower triangular and the latter diagonal. Since the mass matrix is positive-definite, both factors $[\boldsymbol{I}+\boldsymbol{H\phi K}]$ and $\boldsymbol{D}$ are invertible. In particular, since $\boldsymbol{D}$ is diagonal, each of its diagonal elements $\boldsymbol{D}(k)$ is invertible and positive definite. A closed-form operator expression for the inverse of the factor $[\boldsymbol{I}+\boldsymbol{H\phi K}]$ is provided by (4.1c). The factorization in (4.1d) is a closed–form L^*DL factorization of $\boldsymbol{\mathcal{M}}^{-1}$. These operator factorization and inversion results for the mass matrix closely parallel similar results for covariance factorization in estimation theory [1, 6]. The operator expression for $\boldsymbol{\mathcal{M}}^{-1}$ also forms the foundation for $O(\mathcal{N})$ articulated body forward dynamics algorithms [4,7,18]. All of the operators involved in the above mass matrix factorization and inversion are synthesized by spatially recursive algorithms.

Recursive Newton-Euler Factorization

The aim of this subsection is to summarize briefly the essential ideas leading to the Newton-Euler Operator Factorization $\boldsymbol{\mathcal{M}}(\boldsymbol{\theta}) = \boldsymbol{H\phi M\phi^* H}$ of the manipulator mass matrix. While this is done here for a serial chain manipulator, the factorization results apply to a much more general class of complex joint-connected mechanical systems, including tree configurations with flexible links and joints [7].

Consider a serial manipulator with $\mathcal{N}$ rigid links as shown in Figure 1. The links are numbered in increasing order from tip to base. The outer-most link is link 1, and the inner-most link is link $\mathcal{N}$. The overall number of degrees-of-freedom for the manipulator is $\mathcal{N}$. There are two joints attached to the k^{th} link. A coordinate frame $\mathcal{O}_k$ is attached to the inboard joint, and another frame $\mathcal{O}_{k-1}^+$ is attached to the outboard joint. Frame $\mathcal{O}_k$ is also the body frame for the k^{th} link. The k^{th} joint connects the $(k+1)^{st}$ and k^{th} links, and its motion is defined as the motion of frame $\mathcal{O}_k$ with respect to frame $\mathcal{O}_k^+$. When applicable, the free-space motion of a manipulator is modeled by attaching a 6 degree-of-freedom joint between the base link and the inertial frame about which the free-space motion occurs. However, in this paper, without loss of generality and for the sake of notational simplicity, all joints are assumed to be single rotational degree-of-freedom joints with the k^{th} joint coordinate given by $\boldsymbol{\theta}(k)$. Extension to joints with more rotational and translational degrees-of-freedom is easy [5].

The spatial velocity of the k^{th} body frame $\mathcal{O}_k$ is $V(k) = [\omega^*(k), v^*(k)]^* \in \mathbb{R}^6$, where $\omega(k)$ and $v(k)$ are the angular and linear velocities of $\mathcal{O}_k$. With $h(k) \in \mathbb{R}^3$ denoting the k^{th} joint axis vector, $\boldsymbol{H}(k) = [h^*(k), 0] \in \mathbb{R}^1 \times \mathbb{R}^6$ denotes the joint map matrix for the joint, and the relative spatial velocity across the k^{th} joint is $\boldsymbol{H}^*(k)\dot{\boldsymbol{\theta}}(k)$. The spatial force of interaction $f(k)$ across the k^{th} joint is $f(k) = [N^*(k), F^*(k)]^* \in \mathbb{R}^6$, where $N(k)$ and $F(k)$ are the moment and force components respectively. The 6×6 spatial inertia matrix $\boldsymbol{M}(k)$ of the k^{th} link in the coordinate frame $\mathcal{O}_k$ is

$$\boldsymbol{M}(k) = \begin{pmatrix} \boldsymbol{\mathcal{J}}(k) & m(k)\tilde{p}(k) \\ -m(k)\tilde{p}(k) & m(k)\boldsymbol{I}_3 \end{pmatrix}$$

where $m(k)$ is the mass, $p(k) \in \mathbb{R}^3$ is the vector from $\mathcal{O}_k$ to the k^{th} link center of mass, and $\boldsymbol{\mathcal{J}}(k) \in \mathbb{R}^{3\times 3}$ is the rotational inertia of the k^{th} link about $\mathcal{O}_k$. $\boldsymbol{I}_3$ is the 3×3 unit matrix.

The recursive Newton–Euler equations are [1,17]

$$\left\{\begin{array}{lcl} & V(\mathcal{N}+1)= 0; & \alpha(\mathcal{N}+1)= 0 \\ \textbf{for } \boldsymbol{k} & = & \boldsymbol{\mathcal{N}} \cdots \mathbf{1} \\ V(k) & = & \boldsymbol{\phi}^*(k+1,k)V(k+1) + \boldsymbol{H}^*(k)\dot{\boldsymbol{\theta}}(k) \\ \alpha(k) & = & \boldsymbol{\phi}^*(k+1,k)\alpha(k+1)+\boldsymbol{H}^*(k)\ddot{\boldsymbol{\theta}}(k)+a(k) \\ \textbf{end loop} & & \end{array}\right.$$

$$\left\{\begin{array}{lcl} & f(0)=0 & \\ \textbf{for } \boldsymbol{k} & = & \mathbf{1} \cdots \boldsymbol{\mathcal{N}} \\ f(k) & = & \boldsymbol{\phi}(k,k-1)f(k-1)+\boldsymbol{M}(k)\alpha(k)+b(k) \\ \boldsymbol{T}(k) & = & \boldsymbol{H}(k)f(k) \\ \textbf{end loop} & & \end{array}\right. \tag{4.2}$$

where $\boldsymbol{T}(k)$ is the applied moment at joint k. The nonlinear, velocity dependent terms $a(k)$ and $b(k)$ are respectively the Coriolis acceleration and the gyroscopic force terms for the k^{th} link. The transformation operator $\boldsymbol{\phi}(k,k-1)$ between the $\mathcal{O}_{k-1}$ and $\mathcal{O}_k$ frames is

$$\boldsymbol{\phi}(k,k-1) = \begin{pmatrix} \boldsymbol{I}_3 & \tilde{l}(k,k-1) \\ 0 & \boldsymbol{I}_3 \end{pmatrix} \in \mathbb{R}^{6\times 6}$$

where $l(k,k-1)$ is the vector from frame $\mathcal{O}_k$ to frame $\mathcal{O}_{(k-1)}$, and $\tilde{l}(k,k-1) \in \mathbb{R}^{3\times 3}$ is the skew–symmetric matrix associated with the cross-product operation.

The "stacked" notation $\boldsymbol{\theta} = \mathrm{col}\left\{\boldsymbol{\theta}(k)\right\} \in \mathbb{R}^{\mathcal{N}}$ is used to simplify the above recursive Newton-Euler equations. This notation [4] eliminates the arguments k associated with the individual links by defining composite vectors, such as $\boldsymbol{\theta}$, which apply to the entire manipulator system. We define

$$\begin{array}{ll} \boldsymbol{T} = \mathrm{col}\left\{\boldsymbol{T}(k)\right\} \in \mathbb{R}^{\mathcal{N}} & V = \mathrm{col}\left\{V(k)\right\} \in \mathbb{R}^{6\mathcal{N}} \\ f = \mathrm{col}\left\{f(k)\right\} \in \mathbb{R}^{6\mathcal{N}} & \alpha = \mathrm{col}\left\{\alpha(k)\right\} \in \mathbb{R}^{6\mathcal{N}} \\ a = \mathrm{col}\left\{a(k)\right\} \in \mathbb{R}^{6\mathcal{N}} & b = \mathrm{col}\left\{b(k)\right\} \in \mathbb{R}^{6\mathcal{N}} \end{array}$$

In this notation, the equations of motion are [1,6]:

$$V = \boldsymbol{\phi}^*\boldsymbol{H}^*\dot{\boldsymbol{\theta}}; \qquad \alpha = \boldsymbol{\phi}^*[\boldsymbol{H}^*\ddot{\boldsymbol{\theta}} + a] \tag{4.3}$$

$$f = \boldsymbol{\phi}[\boldsymbol{M}\alpha + b]; \qquad \boldsymbol{T} = \boldsymbol{H}f = \boldsymbol{\mathcal{M}}\ddot{\boldsymbol{\theta}} + \boldsymbol{\mathcal{C}} \tag{4.4}$$

where the mass matrix $\mathcal{M}(\theta) = H\phi M\phi H^*$; $\mathcal{C}(\theta,\dot{\theta}) = H\phi[M\phi^* a + b] \in \mathbb{R}^{\mathcal{N}}$ is the Coriolis term; $H = \text{diag}\left\{H(k)\right\} \in \mathbb{R}^{\mathcal{N}\times 6\mathcal{N}}$; $M = \text{diag}\left\{M(k)\right\} \in \mathbb{R}^{6\mathcal{N}\times 6\mathcal{N}}$; and

$$\phi = (I - \mathcal{E}_\phi)^{-1} = \begin{pmatrix} I & 0 & \dots & 0 \\ \phi(2,1) & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \phi(n,1) & \phi(n,2) & \dots & I \end{pmatrix} \in \mathbb{R}^{6\mathcal{N}\times 6\mathcal{N}} \tag{4.5}$$

with $\phi(i,j) = \phi(i,i-1)\cdots\phi(j+1,j)$ for $i > j$. The shift operator $\mathcal{E}_\phi$ is defined as

$$\mathcal{E}_\phi = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ \phi(2,1) & 0 & \dots & 0 & 0 \\ 0 & \phi(3,2) & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \phi(\mathcal{N},\mathcal{N}-1) & 0 \end{pmatrix} \in \mathbb{R}^{6\mathcal{N}\times 6\mathcal{N}} \tag{4.6}$$

Innovations Factorization By Spatial Kalman Filtering

The innovations factorization of the mass matrix is $\mathcal{M} = [I + H\phi K]D[I + H\phi K]^*$, and that of its inverse is $\mathcal{M}^{-1} = [I - H\psi K]^* D^{-1}[I - H\psi K]$. The spatial operators ϕ, K and D embedded in these factorizations are based on spatially recursive filtering and smoothing algorithms [1,4,6]. The following Riccati equation for the articulated body inertia P is a key element of these filtering and smoothing algorithms.

Algorithm 4.1

The articulated body inertia quantities $P(.)$, $D(.)$, $G(.)$, $K(.)$, $\tau(.)$, $\overline{\tau}(.)$, $P^+(.)$ and $\psi(.,.)$ are computed by

$$\left\{ \begin{array}{rcl} & P^+(0) = 0 & \\ \textbf{for } k & = & 1\cdots\mathcal{N} \\ P(k) & = & \phi(k,k-1)P^+(k-1)\phi^*(k,k-1) + M(k) \\ D(k) & = & H(k)P(k)H^*(k) \\ G(k) & = & P(k)H^*(k)D^{-1}(k) \\ K(k+1,k) & = & \phi(k+1,k)G(k) \\ \tau(k) & = & G(k)H(k) \\ \overline{\tau}(k) & = & I - \tau(k) \\ P^+(k) & = & \overline{\tau}(k)P(k) \\ \psi(k+1,k) & = & \phi(k+1,k)\overline{\tau}(k) \\ \textbf{end loop} & & \end{array} \right. \tag{4.7}$$

This algorithm is equivalent to the following spatial operator equation

$$M = P - \mathcal{E}_\psi P \mathcal{E}_\psi^* = P - \mathcal{E}_\phi P \mathcal{E}_\psi^* \tag{4.8}$$

Algorithm 4.1 is the by now classical [4,14] Riccati equation of Kalman filtering. Its solution $P(k)$ is the articulated body inertia [4,18] of the part of the manipulator outboard (toward the tip) of joint k. The operator $\boldsymbol{P}$ is a block-diagonal $6\mathcal{N} \times 6\mathcal{N}$ matrix with its k^{th} diagonal element being $\boldsymbol{P}(k) \in \mathbb{R}^{6\times 6}$. Define also

$$\begin{aligned}
\boldsymbol{D} &= \boldsymbol{HPH}^* \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}} \\
\boldsymbol{G} &= \boldsymbol{PH}^*\boldsymbol{D}^{-1} \in \mathbb{R}^{6\mathcal{N}\times\mathcal{N}} \\
\boldsymbol{K} &= \boldsymbol{\mathcal{E}_\phi G} \in \mathbb{R}^{6\mathcal{N}\times\mathcal{N}} \\
\overline{\boldsymbol{\tau}} &= \boldsymbol{I} - \boldsymbol{GH} \in \mathbb{R}^{6\mathcal{N}\times 6\mathcal{N}} \\
\boldsymbol{\mathcal{E}_\psi} &= \boldsymbol{\mathcal{E}_\phi}\overline{\boldsymbol{\tau}} \in \mathbb{R}^{6\mathcal{N}\times 6\mathcal{N}} \\
\boldsymbol{\psi} &= (\boldsymbol{I} - \boldsymbol{\mathcal{E}_\psi})^{-1} \in \mathbb{R}^{6\mathcal{N}\times 6\mathcal{N}}
\end{aligned} \tag{4.9}$$

The operators $\boldsymbol{D}$,$\boldsymbol{G}$ and $\overline{\boldsymbol{\tau}}$ are all block diagonal. The operators $\boldsymbol{K}$ and $\boldsymbol{\mathcal{E}_\psi}$ are not block-diagonal, but their only nonzero block elements are $\boldsymbol{K}(k,k-1)$'s and $\boldsymbol{\psi}(k,k-1)$'s respectively along the first subdiagonal. The block elements of the lower-block-triangular operator $\boldsymbol{\psi}$ are: $\boldsymbol{\psi}(i,j)$=$\boldsymbol{\psi}(i,i-1)\cdots\ \boldsymbol{\psi}(j+1,j)$ for $i > j$; $\boldsymbol{\psi}(i,j) = \boldsymbol{I}$ for $i=j$; and $\boldsymbol{\psi}(i,j)$= 0 for $i<j$. The structure of the operators $\boldsymbol{\psi}$ and $\boldsymbol{\mathcal{E}_\psi}$ is identical to that of the operators $\boldsymbol{\phi}$ and $\boldsymbol{\mathcal{E}_\phi}$ in (4.5) and (4.6), except that the elements are now $\boldsymbol{\psi}(i,j)$ rather than $\boldsymbol{\phi}(i,j)$.

Key Spatial Operator Identities

Several of the operators above, such as $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$, are related to each other by the following identities discussed in more detail in [4]. These identities will be used later to develop a closed-form expression for the Coriolis term.

Identity 4.2

$$[\boldsymbol{I} - \boldsymbol{H\psi K}]\boldsymbol{H\phi} = \boldsymbol{H\psi} \tag{4.10a}$$

$$\boldsymbol{\phi K}[\boldsymbol{I} - \boldsymbol{H\psi K}] = \boldsymbol{\psi K} \tag{4.10b}$$

$$\boldsymbol{\phi M \phi}^* H^* = [\boldsymbol{I} + \boldsymbol{\phi KH}]\boldsymbol{P\phi}^* H^* \tag{4.10c}$$

$$\boldsymbol{H\phi}\overline{\boldsymbol{\tau}} = \boldsymbol{H\phi\mathcal{E}_\psi} \tag{4.10d}$$

Proof:

$$\boldsymbol{\psi}^{-1} - \boldsymbol{\phi}^{-1} \overset{(4.5,4.9)}{=} \boldsymbol{\mathcal{E}_\phi} - \boldsymbol{\mathcal{E}_\psi} \overset{(4.7)}{=} \boldsymbol{\mathcal{E}_\phi \tau} \overset{(4.7)}{=} \boldsymbol{KH} \tag{4.11}$$

Pre- and post-multiplying this by $\boldsymbol{\psi}$ *and* $\boldsymbol{\phi}$ *respectively implies* $\boldsymbol{\phi} - \boldsymbol{\psi} = \boldsymbol{\psi KH\phi}$, *from which* (4.10a) *follows. Similarly, pre- and post-multiplying* (4.11) *by* $\boldsymbol{\phi}$ *and* $\boldsymbol{\psi}$ *respectively leads to* (4.10b). *Pre- and post-multiplying* (4.8) *by* $\boldsymbol{\phi}$ *and* $\boldsymbol{\phi}^*$ *respectively leads to*

$$\begin{aligned}
\boldsymbol{\phi M \phi}^* &\overset{(4.7)}{=} \boldsymbol{\phi P \phi}^* - \tilde{\boldsymbol{\phi}}\ \overline{\boldsymbol{\tau}}\boldsymbol{P}\tilde{\boldsymbol{\phi}}^* = \boldsymbol{\phi P\phi}^* - \tilde{\boldsymbol{\phi}}\boldsymbol{P}\tilde{\boldsymbol{\phi}}^* + \tilde{\boldsymbol{\phi}}\boldsymbol{\tau P}\tilde{\boldsymbol{\phi}}^* \\
&\overset{(4.7)}{=} \tilde{\boldsymbol{\phi}}\boldsymbol{P} + \boldsymbol{P\phi}^* + \boldsymbol{\phi KHP\phi}^* - \boldsymbol{\phi KHP} = \tilde{\boldsymbol{\phi}}\overline{\boldsymbol{\tau}}\boldsymbol{P} + [\boldsymbol{I} + \boldsymbol{\phi KH}]\boldsymbol{P\phi}^*
\end{aligned}$$

where $\tilde{\boldsymbol{\phi}}$ *is defined as* $\tilde{\boldsymbol{\phi}} = \boldsymbol{\phi}\mathcal{E}_{\boldsymbol{\phi}} = \boldsymbol{\phi} - I$. (4.10c) *follows by post multiplying by* H^* *and noting that*

$$\overline{\boldsymbol{\tau}}\boldsymbol{P}H^* \overset{(4.7)}{=} \boldsymbol{P}\overline{\boldsymbol{\tau}}^* H^* \quad \textit{and} \quad \overline{\boldsymbol{\tau}}^* H^* = 0 \tag{4.12}$$

(4.10d) *is established by*

$$\boldsymbol{H\phi\mathcal{E}_\psi} = \boldsymbol{H}\tilde{\boldsymbol{\phi}}\overline{\boldsymbol{\tau}} = \boldsymbol{H\phi}\overline{\boldsymbol{\tau}} - \boldsymbol{H}\overline{\boldsymbol{\tau}} = \boldsymbol{H\phi}\overline{\boldsymbol{\tau}}$$

Physical Meaning of Spatial Operators

We discuss here, using Table 1 as a summary, the physical meaning of the spatial operators involved in the innovations and Newton-Euler mass matrix factorizations, and in the spatially recursive algorithms that synthesize the spatial operators.

The operator $\boldsymbol{\phi}(k, k-1)$ converts a spatial force at the inboard frame $\mathcal{O}_{k-1}$ and transforms it across the $(k-1)^{th}$ joint and the k^{th} rigid link into a corresponding spatial force at the inboard k^{th} joint frame $\mathcal{O}_k$. Its transpose $\boldsymbol{\phi}^*(k, k-1)$ transforms spatial velocities and accelerations in the opposite direction. Both transformations are rigid in the sense that the joint $k-1$ is kept locked, and the body k to which the operator corresponds is consequently a rigid body. The operator $\boldsymbol{\mathcal{E}_\phi}$ is a shift operator whose elements are all zero, except along its lower sub–diagonal as shown in (4.6). In addition to producing a shift, it rigidly transforms all the forces in the manipulator from the inboard frame of each link to the inboard frame of the next link. Its transpose $\boldsymbol{\mathcal{E}_\phi}^*$ produces a shift and a velocity transfer in the outward direction. The operator $\boldsymbol{H}$ projects spatial forces at the joints into generalized force components along the joint axes. Its transpose $\boldsymbol{H}^*$ converts or "expands" the scalar rotational rates along the joint axes into 6–dimensional relative spatial velocities across the joint.

The articulated body inertia $\boldsymbol{P}$ is the solution to the Riccati equation. Its diagonal element $\boldsymbol{P}(k)$ at joint k is the effective inertia [18] at frame $\mathcal{O}_k$ of the articulated body consisting of links 1 through k. The articulated body inertia captures the "broken bicycle chain" effect. That is, if a bicycle chain is broken and held firmly at its k^{th} link, $\boldsymbol{P}(k)$ is the effective spatial inertia felt at that link. Its value depends upon the chain configuration determined by the angles at the outboard joints. The articulated joint inertia $\boldsymbol{D}(k)$ at joint k is a scalar quantity obtained by projecting the articulated body inertia $\boldsymbol{P}(k)$ along the k^{th} joint axis. The Kalman gain $\boldsymbol{G}$ is computed from the articulated body inertia and appears [1] as a key element in the recursive filtering and smoothing algorithms. Its primary function is to compute the joint articulation operator $\overline{\boldsymbol{\tau}}$ whose diagonal element $\overline{\boldsymbol{\tau}}(k)$ at joint k is used in the Riccati equation of Algorithm 4.1 to remove the scalar rotational inertia about that joint, thereby rendering the resulting body outboard of this joint as an articulated body. The operator $\boldsymbol{\mathcal{E}_\psi}$ is similar to $\boldsymbol{\mathcal{E}_\phi}$, except that it produces "articulated" shifts instead of "rigid" shifts. The operator $\boldsymbol{\psi}$ is a lower-triangular matrix representing an inward spatial Kalman filtering recursion [6]. It is used to propagate forces in an inward direction. In crossing each joint, the articulation operator $\overline{\boldsymbol{\tau}}(k)$ is applied. This is the reason for using the term articulated force transformation to refer to the action of this operator. Its transpose $\boldsymbol{\psi}^*$ is an upper-triangular matrix used to propagate velocities in an outward direction across articulated bodies. The operator $\boldsymbol{\psi}$ differs from the operator $\boldsymbol{\phi}$ in that it takes into account the articulation at the joints which the latter does not.

The Reduced Manipulator $\mathcal{A}_k$ at the k^{th} link

While discussing the articulated body quantities and their physical meaning, a useful notion is that of a *reduced manipulator*. We define a reduced manipulator $\mathcal{A}_k$ at link k to be a manipulator consisting of just links 1 through k and including joint k. Associated with every link in the manipulator is a reduced manipulator. The reduced manipulator $\mathcal{A}_1$ consists of just link 1 and joint 1, while the reduced manipulator $\mathcal{A}_\mathcal{N}$ associated with the $\mathcal{N}^{th}$ link is the whole manipulator itself. In general, the reduced manipulator $\mathcal{A}_{k+1}$ consists of the reduced manipulator $\mathcal{A}_k$ with the $(k+1)^{th}$ link and $(k+1)^{th}$ joint added on to its base. The reduced manipulator $\mathcal{A}_k$ can be regarded as the original manipulator in which all the joints inboard of the k^{th} joint have been locked.

5 The Innovations Factors Diagonalize The Mass Matrix

The innovations factorization in Identity 4.1 leads to a set of diagonal equations of motion. To this end, define the operators $\boldsymbol{m}(\boldsymbol{\theta})$ and $\boldsymbol{\ell}(\boldsymbol{\theta})$ as

$$\boldsymbol{m}(\boldsymbol{\theta}) \stackrel{\triangle}{=} [\boldsymbol{I} + \boldsymbol{H\phi K}]\boldsymbol{D}^{\frac{1}{2}} \qquad \boldsymbol{\ell}(\boldsymbol{\theta}) \stackrel{\triangle}{=} \boldsymbol{m}^{-1}(\boldsymbol{\theta}) = \boldsymbol{D}^{-\frac{1}{2}}[\boldsymbol{I} - \boldsymbol{H\psi K}] \tag{5.1}$$

so that

$$\boldsymbol{\mathcal{M}}(\boldsymbol{\theta}) \stackrel{4.1b}{=} \boldsymbol{m}(\boldsymbol{\theta})\boldsymbol{m}^*(\boldsymbol{\theta}); \qquad \boldsymbol{\mathcal{M}}^{-1}(\boldsymbol{\theta}) \stackrel{4.1d}{=} \boldsymbol{\ell}^*(\boldsymbol{\theta})\boldsymbol{\ell}(\boldsymbol{\theta}) \tag{5.2}$$

The function $\boldsymbol{m}(\boldsymbol{\theta})$ so defined satisfies all of the conditions in Assumption 2, although verifying the condition of differentiability requires the following more careful argument. The operators $\boldsymbol{H}$ and $\boldsymbol{\phi}$ are smooth and differentiable functions of the configuration coordinates, so the only potential trouble-spot is in the differentiability of the articulated body quantities in (4.8), particularly the inverse $\boldsymbol{D}^{-1}$ of the diagonal operator $\boldsymbol{D} = \boldsymbol{HPH}^*$. The diagonal matrix $\boldsymbol{D}$ is always positive definite, invertible and a smooth function of the generalized coordinates. Consequently, $\boldsymbol{D}^{-1}$ is always a smooth and differentiable function of $\boldsymbol{\theta}$. Thus, $\boldsymbol{m} = [\boldsymbol{I} + \boldsymbol{H\phi K}]\boldsymbol{D}^{\frac{1}{2}}$ is also a smooth and differentiable matrix function. Thus, $\boldsymbol{m}(\boldsymbol{\theta})$ satisfies all the conditions in Assumption 2.

The Relative and Total Joint Rates Are Easily Computed From Each Other

The total joint rates $\boldsymbol{\nu}$ are computed from the relative joint rates $\dot{\boldsymbol{\theta}}$ by means of the transformation $\boldsymbol{\nu} = \boldsymbol{m}^*\dot{\boldsymbol{\theta}}$. This transformation is mechanized by an outward recursion from the base of the manipulator to its tip. This outward recursion is specified by the algorithm on the left column of Table 2. The inverse transformation $\dot{\boldsymbol{\theta}} = \boldsymbol{\ell}^*\boldsymbol{\nu}$ is also mechanized by an outward recursion. The right column of Table 2 shows this algorithm.

Similarly, the "new" input variables $\boldsymbol{\epsilon}$ appearing in the diagonalized equation $\dot{\boldsymbol{\nu}} + \boldsymbol{\mathcal{C}}(\boldsymbol{\nu}, \boldsymbol{\theta}) = \boldsymbol{\epsilon}$ are obtained from the "old" inputs $\boldsymbol{T}$ by the transformation $\boldsymbol{\epsilon} = \boldsymbol{\ell T}$. This is mechanized by the inward, tip-to-base recursion specified on the left column of Table 3. The inverse operation $\boldsymbol{T} = \boldsymbol{m\epsilon}$ from the new variables $\boldsymbol{\epsilon}$ to the old variables $\boldsymbol{T}$ is also performed recursively in an outward direction, as specified by the algorithm in the right column of Table 3.

It is relatively easy therefore to go back and forth between the "old" variables $\dot{\boldsymbol{\theta}}$ and $\boldsymbol{T}$ in traditional robot dynamics and the "new" variables $\boldsymbol{\nu}$ and $\boldsymbol{\epsilon}$ in the diagonalized equations of this paper. The two mutually reciprocal outward recursions in Table 2 govern the relationships between the new and old velocities. The two mutually reciprocal inward recursions in the Table 3 govern the relationships between the new and old input forces. The term "mutually reciprocal" indicates that the corresponding spatial operations are mathematical inverses of each other. Each of the above four recursions represents an $O(\mathcal{N})$ computational algorithm, in the sense that the number of required arithmetical operations increases only linearly with the number of degrees of freedom.

Physical Interpretation of the Total Joint Velocities

The total joint velocities $\boldsymbol{\nu}$ can be obtained from the joint-angle velocities by means of the recursion on the left column of Table 2. There is a physical interpretation to this. Observe from Table 2 that

$$\boldsymbol{D}^{-\frac{1}{2}}(k)\boldsymbol{\nu}(k) = \dot{\boldsymbol{\theta}}(k) + \delta(k), \quad \text{where} \quad \delta(k) \stackrel{\triangle}{=} \boldsymbol{G}^*(k)V^+(k) \tag{5.3}$$

in which $V^+(k)$ is the spatial velocity of frame $\mathcal{O}_k^+$ which is immediately adjacent to and on the inboard side of the k^{th} joint. This spatial velocity is due to the relative velocities $\dot{\boldsymbol{\theta}}(j)$ at all of the joints inboard of joint k. The spatial velocity $V^{+}(k)$ represents the spatial velocity of the "base–body" of the k^{th} reduced manipulator $\mathcal{A}_k$. The quantity $\boldsymbol{D}^{\frac{1}{2}}(k)$ is a normalizing factor which is used so that the kinetic energy is not only diagonalized but normalized as in (1.2).

(5.3) states that the total joint rate $\boldsymbol{D}^{-\frac{1}{2}}(k)\boldsymbol{\nu}(k)$ at joint k is the sum of two angular rates. One of these is the relative joint velocity $\dot{\boldsymbol{\theta}}(k)$ at joint k between link k and the next link $k+1$, which is the joint velocity at the base link of the reduced manipulator $\mathcal{A}_k$. The second angular rate given by $\delta(k)$ represents an additional term due to the non–zero spatial velocity $V^{+}(k)$ of the "base–body" of $\mathcal{A}_k$. When link $(k+1)$ is at rest, the additional term $\delta(k)$ is zero, and $\boldsymbol{D}^{-\frac{1}{2}}(k)\boldsymbol{\nu}(k)$ equals the commonly used joint relative rate $\dot{\boldsymbol{\theta}}(k)$. The correction term $\delta(k)$ depends on the articulated body inertia quantities $\boldsymbol{P}(k)$ and $\boldsymbol{D}(k)$. It compensates for the joint motion induced in all the outboard joints by the motion of the "base–body" of the redundant manipulator at joint k.

Physical Interpretation of the New Generalized Forces

The input variables $\boldsymbol{\epsilon}$ in the new equations of motion also have a nice physical interpretation. This can be seen from the relationship

$$\boldsymbol{T}(k) = \boldsymbol{D}^{\frac{1}{2}}(k)\boldsymbol{\epsilon}(k) + \boldsymbol{H}(k)z(k) \tag{5.4}$$

One way to interpret this relationship is to observe that the applied moment $\boldsymbol{T}(k)$ at joint k is the sum of two terms. The first term $\boldsymbol{D}^{\frac{1}{2}}(k)\boldsymbol{\epsilon}(k)$ is a working joint moment in the sense that it directly enters the diagonalized equation $\dot{\boldsymbol{\nu}} + \boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu}) = \boldsymbol{\epsilon}$ and causes the "acceleration" term $\dot{\boldsymbol{\nu}}$ to either increase or decrease. The second component $H(k)z(k)$ depends only upon, and compensates for all, the outboard applied moments $\boldsymbol{T}(1), \cdots, \boldsymbol{T}(k-1)$. A point worth noting here is that $\boldsymbol{\epsilon}(k)$ depends only on quantities associated with the reduced manipulator $\mathcal{A}_k$.

Extension of the Cross Product Operation to Spatial Vectors

The cross–product $x \times y$ of a pair of 3–dimensional vectors x and y can also be written as $\tilde{x}y$ where $\tilde{x} \in \mathbb{R}^{3\times 3}$ is the appropriate skew–symmetric matrix. We introduce here a "cross–product" operator for 6–dimensional spatial quantities as follows. Let $X = [a^*, b^*]^*$ and $Y = [c^*, b^*]^*$ be two arbitrary spatial vectors where a, b, c, d are 3–dimensional vectors. Then, the "cross–product" operation $X \times Y$, is defined as

$$X \times Y \triangleq \begin{pmatrix} a \\ b \end{pmatrix} \times \begin{pmatrix} c \\ d \end{pmatrix} = \tilde{X}Y = \begin{pmatrix} \tilde{a}c \\ \tilde{b}c + \tilde{a}d \end{pmatrix}, \quad \text{where} \quad \tilde{X} \triangleq \begin{pmatrix} \tilde{a} & 0 \\ \tilde{b} & \tilde{a} \end{pmatrix} \in \mathbb{R}^{6\times 6} \tag{5.5}$$

The spatial cross–product operation is anti–symmetric, i.e. $X \times Y = -Y \times X$ and satisfies the identity $X^*(Z \times Y) = -Z^*(X \times Y)$. While the operation "$\times$" is anti–symmetric for spatial vectors, the matrix $\tilde{X}$ is *not* skew–symmetric, i.e $\tilde{X} \neq -\tilde{X}^*$ except in the case where the lower half of X is zero. Given spatial vectors $X(k)$, and the vector $X = \text{col}\left\{X(k)\right\}_{k=1}^{\mathcal{N}} \in \mathbb{R}^{6\mathcal{N}}$, we define

$$\tilde{X} \triangleq \text{diag}\left\{\tilde{X}(k)\right\} \in \mathbb{R}^{6\mathcal{N}\times 6\mathcal{N}} \qquad \text{so that} \qquad \tilde{X}Y = \text{col}\left\{\tilde{X}(k)Y(k)\right\}_{k=1}^{\mathcal{N}} \in \mathbb{R}^{6\mathcal{N}} \tag{5.6}$$

6 Mass Matrix Derivatives in The Coriolis Term

The Coriolis term $\mathcal{C}(\boldsymbol{\theta},\boldsymbol{\nu})=\boldsymbol{\ell}(\dot{\boldsymbol{m}}\boldsymbol{\nu}-\frac{1}{2}\dot{\boldsymbol{\theta}}^*\mathcal{M}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}})$ is one of the key elements in the diagonalized equations of motion $\dot{\boldsymbol{\nu}}+\mathcal{C}(\boldsymbol{\theta},\boldsymbol{\nu})=\boldsymbol{\epsilon}$. There are two key computations in this term:

- The inertial time derivative $\dot{\boldsymbol{m}}$ of the mass matrix factor $\boldsymbol{m}$.
- The first-order derivative or "sensitivity" $\mathcal{M}_{\boldsymbol{\theta}}$ of the mass matrix $\mathcal{M}$ with respect to the joint angles $\boldsymbol{\theta}$.

This section summarizes key results regarding the differentiation of spatial operators. With these results, the mass matrix derivatives in the Coriolis term can be computed with relative ease.

Time Derivatives of Key Spatial Operators

The inertial time derivative $\dot{x}$ of a quantity x is taken with respect to an inertially fixed frame. The local time derivative $\mathring{x}(k)$ of a quantity associated with body k is taken with respect to the k^{th} body frame $\mathcal{O}_k$ and takes into account only the internal changes within the reduced manipulator $\mathcal{A}_{k-1}$, caused by motion at the joints $1,\cdots,k-1$. The inertial derivative on the other hand takes into account the motion of the remaining inboard joints $k,\cdots,\mathcal{N}$ as well. The inertial time derivative $\dot{x}(k)$ of an arbitrary 6-dimensional spatial vector $x(k)$ attached to body k is related to its local time derivative $\mathring{x}(k)$ by:

$$\dot{x}(k)=\mathring{x}(k)+\tilde{\boldsymbol{\Omega}}(k)x(k) \tag{6.1}$$

where $\tilde{\boldsymbol{\Omega}}(k)$ is the spatial cross product matrix associated with the spatial vector $\boldsymbol{\Omega}(k)$, where $\boldsymbol{\Omega}k$ is defined as:

$$\boldsymbol{\Omega}(k)\triangleq\begin{pmatrix}\omega(k)\\ 0\end{pmatrix} \tag{6.2}$$

Thus the inertial time derivative of the various operator quantities at the k^{th} link consist of two components: one component arises purely due to the the motion of joints $1,\cdots,k-1$ outboard of link k; and a second term due to the non–zero angular velocity $\omega(k)$ of the frame $\mathcal{O}_k$ with respect to the inertial frame. For quantities such as $\boldsymbol{\phi}(k,j)$ that involve more than one link, the local time dervative $\mathring{\boldsymbol{\phi}}(k,j)$ is defined with respect to the body frame which is inboard of the other. Thus, if $k>j$ in this example, then the local time derivative of $\boldsymbol{\phi}(k,j)$ is defined with respect to the reduced manipulator $\mathcal{A}_k$.

Define also the quantities $\boldsymbol{\Omega}_\delta(k)\in\mathbb{R}^6$ and $\boldsymbol{\Omega}_\delta\in\mathbb{R}^{6\mathcal{N}}$ as follows:

$$\boldsymbol{\Omega}_\delta(k)\triangleq\boldsymbol{\Omega}(k)-\boldsymbol{\Omega}(k+1)=\boldsymbol{H}^*(k)\dot{\boldsymbol{\theta}}(k),\quad\text{and}\quad\boldsymbol{\Omega}_\delta\triangleq\text{col}\left\{\boldsymbol{\Omega}_\delta(k)\right\} \tag{6.3}$$

Table 4 summarizes some of the key expressions for the derivatives of various operator quantities. It is easy to establish this table. In general, in going from top to bottom, each row follows from the previous ones by using the chain-rule of differentiation. Only the local derivatives in Table 4 are discussed here. The inertial derivatives in the table can be established similarly.

To establish the first row, observe that $\overset{\circ}{\boldsymbol{\phi}}(k+1,k) = \boldsymbol{\phi}(k+1,k)\tilde{\boldsymbol{\Omega}}_\delta(k)$. The second row follows by differentiating $(\boldsymbol{I}-\boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}})\boldsymbol{\phi} = \boldsymbol{I}$. Rows 3 and 4 follow because the operators $\boldsymbol{H}(k)$ and $\boldsymbol{M}(k)$ are rigidly attached to body k. Row 5 follows from Row 3. Row 6 follows because $\boldsymbol{GD} = \boldsymbol{PH}^*$. Hence, $\overset{\circ}{\boldsymbol{G}}\boldsymbol{D} + \boldsymbol{G}\overset{\circ}{\boldsymbol{D}} = \overset{\circ}{\boldsymbol{P}}\boldsymbol{H}^*$. Use $\overset{\circ}{\boldsymbol{D}} = \boldsymbol{H}\overset{\circ}{\boldsymbol{P}}\boldsymbol{H}^*$ and rearrange terms. Rows 7 follows from Row 6. Rows 1 and 7 imply Row 8.

The local time derivative $\overset{\circ}{\boldsymbol{P}}(k)$ of the articulated body inertia $\boldsymbol{P}(k)$ is a key quantity required to evaluate the time derivatives of several of the spatial operators in Table 4, as well as the term $\dot{\boldsymbol{m}}$ in the Coriolis force $\boldsymbol{\mathcal{C}}(\boldsymbol{\nu},\boldsymbol{\theta})=\boldsymbol{\ell}(\dot{\boldsymbol{m}}\boldsymbol{\nu} - \frac{1}{2}\dot{\boldsymbol{\theta}}^*\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}})$. Because of this, the local time derivative $\overset{\circ}{\boldsymbol{P}}$ requires special consideration.

Local Time Derivative of the Articulated Inertia

This section discusses the local time derivative $\overset{\circ}{\boldsymbol{P}}(k)$ of the articulated body inertia $\boldsymbol{P}(k)$ and the closely related quantity $\dot{\boldsymbol{\lambda}}(k)$, which is the inertial time derivative of $\boldsymbol{P}(k)$ with respect to the coordinates of the reduced manipulator $\mathcal{A}_k$ alone. The following algorithm is established readily by local-time differentiation of the Riccati equation and use of Table 4.

Algorithm 6.1 *The local time-derivative $\overset{\circ}{\boldsymbol{P}}$ of the articulated inertia $\boldsymbol{P}$ and the related quantity $\dot{\boldsymbol{\lambda}}$ satisfy the operator equation:*

$$\dot{\boldsymbol{\lambda}} = \overset{\circ}{\boldsymbol{P}} + \tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{P} - \boldsymbol{P}\tilde{\boldsymbol{\Omega}}_\delta, \quad \textit{and} \quad \overset{\circ}{\boldsymbol{P}} = \boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}}\dot{\boldsymbol{\lambda}}\boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}}^* \tag{6.4}$$

corresponding to the recursive algorithm:

$$\left\{\begin{array}{rcl} & \dot{\boldsymbol{\lambda}}(0) = 0 & \\ \textbf{for } \boldsymbol{k} & = & \mathbf{1}\cdots\boldsymbol{\mathcal{N}} \\ \overset{\circ}{\boldsymbol{P}}(k) & = & \boldsymbol{\psi}(k,k-1)\dot{\boldsymbol{\lambda}}(k-1)\boldsymbol{\psi}(k,k-1) \\ \dot{\boldsymbol{\lambda}}(k) & = & \overset{\circ}{\boldsymbol{P}}(k) + \tilde{\boldsymbol{\Omega}}_\delta(k)\boldsymbol{P}(k) - \boldsymbol{P}(k)\tilde{\boldsymbol{\Omega}}_\delta(k) \\ \textbf{end loop} & & \end{array}\right. \tag{6.5}$$

The above algorithm consists of an inward recursion from the manipulator tip to its base. It is a "sensitivity" equation corresponding to the articulated body inertia Riccati equation. The algorithm computes $\dot{\boldsymbol{\lambda}}$ and the local time derivative $\overset{\circ}{\boldsymbol{P}}$ of the articulated body inertia $\boldsymbol{P}$, in terms of the articulated body inertia $\boldsymbol{P}$ itself. The recursion is linear, with the term $\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{P} - \boldsymbol{P}\tilde{\boldsymbol{\Omega}}_\delta$ being an input. For each joint k, this term reflects the change in the articulated body inertia due to the rotation $\tilde{\boldsymbol{\Omega}}_\delta(k)$. Because the algorithm is an inward recursion, the time derivative $\dot{\boldsymbol{\lambda}}(k)$ at a joint k depends only on the rotation at the joints of the reduced manipulator $\mathcal{A}_k$. The time derivative $\dot{\boldsymbol{\lambda}}$ does not depend on the joints $k+1,\cdots,\mathcal{N}$ on the inward path toward the manipulator base. The inertial time derivative of the articulated body inertia $\dot{\boldsymbol{P}}$ satisfies the following relationship:

$$\dot{\boldsymbol{P}} = \overset{\circ}{\boldsymbol{P}} + \tilde{\boldsymbol{\Omega}}\boldsymbol{P} - \boldsymbol{P}\tilde{\boldsymbol{\Omega}} \tag{6.6}$$

Local Time Derivative of the ϕ Operator

Lemma 6.1

$$\overset{\circ}{\boldsymbol{\phi}} \;=\; \tilde{V}^*\boldsymbol{\phi} - \boldsymbol{\phi}\tilde{V}^* \tag{6.7}$$

which implies

$$\boldsymbol{H}[\tilde{V}^*\boldsymbol{\phi} - \boldsymbol{\phi}\tilde{V}^*] \quad = \quad \boldsymbol{H}\boldsymbol{\phi}\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{\phi} \tag{6.8}$$

Proof: *Where convenient, we use* $A^\times$ *in place of* $\tilde{A}$*. For any spatial vector* $X \in \mathbb{R}^6$ *and any* $l \in \mathbb{R}^3$,

$$[\boldsymbol{\phi}^*(l)X]^\times = \boldsymbol{\phi}^*(l)\tilde{X}\boldsymbol{\phi}^{-*}(l) \quad \text{where} \quad \boldsymbol{\phi}(l) \stackrel{\triangle}{=} \begin{pmatrix} \boldsymbol{I}_3 & l \\ 0 & \boldsymbol{I}_3 \end{pmatrix} \tag{6.9}$$

Applying (6.9) *to* $V^+(k) = \boldsymbol{\phi}^*(k+1,k)V(k+1)$,

$$\begin{aligned} \tilde{V}^+(k) = [\boldsymbol{\phi}^*(k+1,k)V(k+1)]^\times &= \boldsymbol{\phi}^*(k+1,k)\tilde{V}(k+1)\boldsymbol{\phi}^{-*}(k+1,k) \\ \Longrightarrow \tilde{V}^+(k)\boldsymbol{\phi}^*(k+1,k) &= \boldsymbol{\phi}^*(k+1,k)\tilde{V}(k+1) \end{aligned} \tag{6.10}$$

Using $V(k) = V^+(k) + H^*\dot{\boldsymbol{\theta}}(k)$, (6.10) *can be recast as* $\boldsymbol{\mathcal{E}}^*_{\boldsymbol{\phi}}\tilde{V} = \tilde{V}^+\boldsymbol{\mathcal{E}}^*_{\boldsymbol{\phi}} = \tilde{V}\boldsymbol{\mathcal{E}}^*_{\boldsymbol{\phi}} - \tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{\mathcal{E}}^*_{\boldsymbol{\phi}}$*. Thus,*

$$-\boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}\tilde{V}^* \quad = \quad -\tilde{V}^*\boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}} + \boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}\tilde{\boldsymbol{\Omega}}_\delta$$

To establish Eq. (6.7), pre- and post-multiply this by $\boldsymbol{\phi}$ *and recall from Table 4 that* $\mathring{\boldsymbol{\phi}} = \boldsymbol{\phi}\mathcal{E}_{\boldsymbol{\phi}}\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{\phi}$*. To establish Eq. (6.8), pre-multiply Eq. (6.7) by* $\boldsymbol{H}$ *and recall that* $\boldsymbol{\phi}\mathcal{E}_{\boldsymbol{\phi}} = \boldsymbol{\phi} - I$ *and* $\boldsymbol{H}\tilde{\boldsymbol{\Omega}}_\delta = 0$*.* ∎

Time Derivative of the Mass Matrix Innovations Factor

Identity 6.1 *The* **local** *and* **inertial** *time derivatives of the mass matrix factor* $\boldsymbol{m}$ *are equal to each other and are*

$$\dot{\boldsymbol{m}} = \mathring{\boldsymbol{m}} = \boldsymbol{H}\boldsymbol{\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta\tilde{\boldsymbol{\phi}}P + \frac{1}{2}(\boldsymbol{I}+\overline{\boldsymbol{\tau}})\dot{\boldsymbol{\lambda}}\right]\boldsymbol{H}^*\boldsymbol{D}^{-\frac{1}{2}} \tag{6.11}$$

Proof: Note that $\boldsymbol{m} = [\boldsymbol{I} + \boldsymbol{H}\boldsymbol{\phi}\boldsymbol{K}]\mathrm{D}^{\frac{1}{2}} = [\boldsymbol{I} + \boldsymbol{H}\tilde{\boldsymbol{\phi}}\boldsymbol{G}]\boldsymbol{D}^{\frac{1}{2}} = [\boldsymbol{I} - \boldsymbol{H}\boldsymbol{G} + \boldsymbol{H}\boldsymbol{\phi}\boldsymbol{G}]\boldsymbol{D}^{\frac{1}{2}} = \boldsymbol{H}\boldsymbol{\phi}\boldsymbol{G}\boldsymbol{D}^{\frac{1}{2}}$. Use items No. 2, 3, 5 and 6 in Table 3 to show that local and inertial derivatives are the same. Use either the left or the right column in Table 4 and get the same answer. ∎

For later convenience, observe also the following additional identity.

Identity 6.2

$$\dot{\boldsymbol{m}}\boldsymbol{\nu} = \boldsymbol{H}\boldsymbol{\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{\phi}\boldsymbol{K}\boldsymbol{H}\boldsymbol{P} + \frac{1}{2}(\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{P} - \boldsymbol{P}\tilde{\boldsymbol{\Omega}}_\delta + \boldsymbol{\mathcal{E}}_\psi\dot{\boldsymbol{\lambda}} - \dot{\boldsymbol{\lambda}}\boldsymbol{\mathcal{E}}^*_\psi)\right]V \tag{6.12}$$

Proof:

$$\begin{aligned}
\dot{m}\boldsymbol{\nu} &= \boldsymbol{H\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta\tilde{\boldsymbol{\phi}}\boldsymbol{P} + \frac{1}{2}(I+\overline{\boldsymbol{\tau}})\dot{\boldsymbol{\lambda}}\right]\boldsymbol{H}^*\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{\nu} \\
&\overset{(5.1)}{=} \boldsymbol{H\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta\tilde{\boldsymbol{\phi}}\boldsymbol{P} + \frac{1}{2}(I+\overline{\boldsymbol{\tau}})\dot{\boldsymbol{\lambda}}\right]\boldsymbol{H}^*\boldsymbol{G}^*\boldsymbol{\phi}^*H^*\dot{\boldsymbol{\theta}} \\
&\overset{(4.3)}{=} \boldsymbol{H\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta\tilde{\boldsymbol{\phi}}\boldsymbol{P} + \frac{1}{2}(I+\overline{\boldsymbol{\tau}})\dot{\boldsymbol{\lambda}}\right]\boldsymbol{H}^*\boldsymbol{G}^*V \\
&\overset{(4.7)}{=} \boldsymbol{H\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{\phi}\boldsymbol{KHP} + \frac{1}{2}(I+\overline{\boldsymbol{\tau}})\dot{\boldsymbol{\lambda}}\boldsymbol{\tau}^*\right]V \\
&\overset{(4.10d)}{=} \boldsymbol{H\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{\phi}\boldsymbol{KHP} + \frac{1}{2}(I+\boldsymbol{\mathcal{E}}_\psi)\dot{\boldsymbol{\lambda}}(I-\boldsymbol{\mathcal{E}}_\psi^*)\right]V \\
&= \boldsymbol{H\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{\phi}\boldsymbol{KHP} + \frac{1}{2}(\dot{\boldsymbol{\lambda}} - \boldsymbol{\mathcal{E}}_\psi\dot{\boldsymbol{\lambda}}\boldsymbol{\mathcal{E}}_\psi^* + \boldsymbol{\mathcal{E}}_\psi\dot{\boldsymbol{\lambda}} - \dot{\boldsymbol{\lambda}}\boldsymbol{\mathcal{E}}_\psi^*)\right]V \\
&\overset{(6.4)}{=} \boldsymbol{H\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{\phi}\boldsymbol{KHP} + \frac{1}{2}(\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{P} - \boldsymbol{P}\tilde{\boldsymbol{\Omega}}_\delta + \boldsymbol{\mathcal{E}}_\psi\dot{\boldsymbol{\lambda}} - \dot{\boldsymbol{\lambda}}\boldsymbol{\mathcal{E}}_\psi^*)\right]V \qquad (6.13)
\end{aligned}$$

∎

Closed-Form Mass Matrix Sensitivity $\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}_i}$ and $\dot{\boldsymbol{\theta}}^*\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}}$

Identity 6.3

$$\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}_i} = \boldsymbol{H\phi}\left[\mathbb{H}_\delta^i\boldsymbol{\phi}\boldsymbol{M} - \boldsymbol{M}\boldsymbol{\phi}^*\mathbb{H}_\delta^i\right]\boldsymbol{\phi}^*\boldsymbol{H}^* \qquad (6.14)$$

There is an important new quantity in this result, and it has a simple physical interpretation. The matrix $\mathbb{H}_\delta^i$ is the $6\mathcal{N}\times 6\mathcal{N}$ matrix whose elements are all zero, except for a single 6×6 block $\tilde{\boldsymbol{H}}(i)$ at the i^{th} location on the diagonal. The index i corresponds to the joint-angle $\boldsymbol{\theta}_i$ with respect to which the sensitivity $\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}_i}$ is being taken. The non-zero block-diagonal element $\tilde{\boldsymbol{H}}(i)$ is obtained as follows by appropriately rearranging the joint axis unit vector $h(i)$ to form the cross-product-like operation

$$\tilde{\boldsymbol{H}}(i) = \begin{pmatrix} \tilde{h}(i) & 0 \\ 0 & \tilde{h}(i) \end{pmatrix} \qquad (6.15)$$

The above formula in Eq. (6.14) is closed-form, in the sense that it explicitly computes the mass matrix sensitivity in terms of the operators $\boldsymbol{\phi}$, $\boldsymbol{M}$, and $\boldsymbol{H}$ appearing in the mass matrix itself. That the formula is closed-form is of extreme importance, because it implies that the mass matrix derivatives can be easily computed using operations and spatially recursive algorithms similar to those used to compute the mass matrix itself. As described later, this allows development of simple closed-form expressions and recursive algorithms to evaluate the Coriolis term $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu})$ in the diagonalized equations of motion.

Identity 6.3 is established by use of rows 2, 3, and 4 in Table 4, together with the classical chain-rule of differentiation. In computing the sensitivity $\mathcal{M}_{\boldsymbol{\theta}_i}$ with respect to the i^{th} joint angle $\boldsymbol{\theta}_i$, the corresponding time derivative in Table 4 is computed assuming that all of the other angles are locked. Use of the local derivatives in Table 4, instead of the inertial derivatives, leads to a simpler derivation of Identity 6.3.

Identity 6.3 leads to the following expression for the term $\dot{\boldsymbol{\theta}}^* \boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}} \dot{\boldsymbol{\theta}}$ in the Coriolis forces vector.

Lemma 6.2

$$\begin{aligned} \dot{\boldsymbol{\theta}}^* \boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}} \dot{\boldsymbol{\theta}} &= 2\boldsymbol{H}\tilde{V}^* \boldsymbol{\phi} \boldsymbol{M} V && (6.16a) \\ &= 2\boldsymbol{H}\boldsymbol{\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta \boldsymbol{\phi} + \tilde{V}^*\right] \boldsymbol{M} V && (6.16b) \\ &= 2\boldsymbol{H}\boldsymbol{\phi}\left[\tilde{\boldsymbol{\Omega}}_\delta (\boldsymbol{I} + \boldsymbol{\phi}\boldsymbol{K}\boldsymbol{H})\boldsymbol{P} + \tilde{V}^*\right] \boldsymbol{M} V && (6.16c) \end{aligned}$$

Proof: *From Identity 6.3,*

$$\text{col}\left\{\dot{\boldsymbol{\theta}}^* \boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}_i} \dot{\boldsymbol{\theta}}\right\} \overset{(4.3)}{=} 2\,\text{col}\left\{V^* \mathbb{H}^i_\delta \boldsymbol{\phi} \boldsymbol{M} V\right\} = 2\,\text{diag}\left\{V^*(k)\mathbb{H}(k)\right\} \boldsymbol{\phi} \boldsymbol{M} V \quad (6.17)$$

Since $(-X^*\tilde{Y}^* = Y^*\tilde{X}^* \ \forall X, Y \in \mathbb{R}^6)$,

$$\text{diag}\left\{V^*(k)\mathbb{H}(k)\right\} = \text{diag}\left\{\boldsymbol{H}(k)\tilde{V}^*(k)\right\} = \boldsymbol{H}\tilde{V}^* \quad (6.18)$$

Substituting this into (6.17) *leads to* (6.16a). (6.16b) *follows from the direct use of* (6.8) *in* (6.16a). *The use of the expression for* V *in* (4.3) *along with* (4.10c) *leads to* (6.16). ∎

7 Closed-Form Expression for the Coriolis Forces $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu})$

Identity 7.1

$$\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu}) = \frac{1}{2}\boldsymbol{D}^{-\frac{1}{2}} \boldsymbol{H}\boldsymbol{\psi}\left[\boldsymbol{\mathcal{E}}_{\psi}\dot{\boldsymbol{\lambda}} - \dot{\boldsymbol{\lambda}}\boldsymbol{\mathcal{E}}^*_{\boldsymbol{\psi}} - \tilde{\boldsymbol{\Omega}}_\delta \boldsymbol{P} - \boldsymbol{P}\tilde{\boldsymbol{\Omega}}_\delta - 2\tilde{V}^*\boldsymbol{M}\right] V \quad (7.1)$$

where $\boldsymbol{V} = \boldsymbol{\psi}^* \boldsymbol{H}^* \boldsymbol{D}^{-\frac{1}{2}} \boldsymbol{\nu}$ *is the composite vector of spatial velocities.*

Proof: Combine Lemma 3.1, (6.12), and (6.16). ∎

Identity 7.1 is a breakthrough. It explicitly evaluates in terms of relatively simple quantities, the very complicated quantity $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu}) = \boldsymbol{\ell}(\dot{\boldsymbol{m}} - \frac{1}{2}\dot{\boldsymbol{\theta}}^* \boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}} \dot{\boldsymbol{\theta}})$ which depends on various derivatives of the system mass matrix. Furthermore, Algorithm 7.1 below computes this term recursively.

Inwardly Recursive Algorithm to Compute $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta}, \boldsymbol{\nu})$

Algorithm 7.1

$$
\left\{
\begin{array}{rcl}
 & & \dot{\boldsymbol{\lambda}}(0) = 0, \;\; y(0) = 0 \\
\textbf{for } \boldsymbol{k} & = & \mathbf{1} \cdots \boldsymbol{\mathcal{N}} \\
X(k) & = & \tilde{\boldsymbol{\Omega}}_\delta(k)\boldsymbol{P}(k) \\
Y(k,k-1) & = & \boldsymbol{\psi}(k,k-1)\dot{\boldsymbol{\lambda}}(k-1) \\
\dot{\boldsymbol{\lambda}}(k) & = & Y(k,k-1)\boldsymbol{\psi}^*(k,k-1)+X(k)+X^*(k) \\
y(k) & = & \boldsymbol{\psi}(k,k-1)y(k-1) - [2\tilde{V}^*(k)\boldsymbol{M}(k)+X(k)-X^*(k)]V(k)+ \\
 & & \quad Y(k,k-1)V(k-1) - \dot{\boldsymbol{\lambda}}(k)\overline{\boldsymbol{\tau}}^*(k)V^+(k) \\
\boldsymbol{\mathcal{C}}(k) & = & \frac{1}{2}D^{-\frac{1}{2}}(k)\boldsymbol{H}(k)y(k) \\
\textbf{end loop} & &
\end{array}
\right.
\tag{7.2}
$$

The above algorithm is in essence a recursive implementation of (7.1). It proceeds from tip-to-base and is of $O(\mathcal{N})$ computational complexity. It assumes that the spatial velocities V have already been computed using Table 2 for example. Similarly, the articulated quantities are either previously or concurrently computed using (4.7).

Coriolis Force Does No Work

The Coriolis term $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu})$ is orthogonal to the generalized velocities $\boldsymbol{\nu}$ and therefore does no mechanical work.

Lemma 7.1

$$\boldsymbol{\nu}^*\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu}) = 0 \tag{7.3}$$

Proof: *Observe that* $\boldsymbol{\mathcal{M}} = \boldsymbol{m}\boldsymbol{m}^*$ *implies that* $\dot{\boldsymbol{\theta}}^*\boldsymbol{\mathcal{M}}_{\boldsymbol{\theta}}\dot{\boldsymbol{\theta}} = 2\,\mathrm{col}\left\{\boldsymbol{\nu}^*\boldsymbol{m}^*_{\boldsymbol{\theta}_i}\dot{\boldsymbol{\theta}}\right\}$. *Consequently,* $\boldsymbol{\nu}^*\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu})$ $= \dot{\boldsymbol{\theta}}^*\left(\dot{\boldsymbol{m}}\boldsymbol{\nu} - \mathrm{col}\left\{\boldsymbol{\nu}^* m^*_{\boldsymbol{\theta}_i}\dot{\boldsymbol{\theta}}\right\}\right) = \dot{\boldsymbol{\theta}}^*\dot{\boldsymbol{m}}\boldsymbol{\nu} - \sum_{i=1}^{\mathcal{N}}\dot{\boldsymbol{\theta}}(i)\boldsymbol{\nu}^* m^*_{\boldsymbol{\theta}_i}\dot{\boldsymbol{\theta}} = \dot{\boldsymbol{\theta}}^*\dot{\boldsymbol{\theta}}\boldsymbol{\nu} - \boldsymbol{\nu}^*\dot{\boldsymbol{m}}^*\dot{\boldsymbol{\theta}} = 0.$ ∎

A similar orthogonality condition can be obtained using the explicit expression for the Coriolis forces vector $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu})$ in Identity 7.1:

$$\boldsymbol{\nu}^*\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu}) = \frac{1}{2}V^*\left[\boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}}\dot{\boldsymbol{\lambda}} - \dot{\boldsymbol{\lambda}}\boldsymbol{\mathcal{E}}^*_{\boldsymbol{\psi}} - \tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{P} - \boldsymbol{P}\tilde{\boldsymbol{\Omega}}_\delta - 2\tilde{V}^*\boldsymbol{M}\right]V \tag{7.4}$$

$$= \frac{1}{2}V^*\left[\boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}}\dot{\boldsymbol{\lambda}} - \dot{\boldsymbol{\lambda}}\boldsymbol{\mathcal{E}}^*_{\boldsymbol{\psi}} - \tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{P} - \boldsymbol{P}\tilde{\boldsymbol{\Omega}}_\delta\right]V = 0 \tag{7.5}$$

Since the matrix expression in the middle is skew–symmetric, the overall expression is zero.

The orthogonality of the nonlinear Coriolis forces is similar to the orthogonality condition $\omega^*[\omega \times \boldsymbol{\mathcal{J}}\omega] = 0$ of the gyroscopic force term in the equations of motion for a single rigid body rotating with angular velocity ω. In contrast, the corresponding Coriolis forces term $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}})$ in the regular equations of motion in (2.1) does work, i.e., $\dot{\boldsymbol{\theta}}^*\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) \neq 0$.

Rate of change of the kinetic energy

The non-working nature of the Coriolis forces has an interesting implication. Recall that the kinetic energy of the system is $\boldsymbol{\mathcal{K}}(\boldsymbol{\theta},\boldsymbol{\nu}) = \frac{1}{2}\boldsymbol{\nu}^*\boldsymbol{\nu}$.

Lemma 7.2 *The rate of change of the kinetic energy is the dot product of the generalized forces and generalized velocities*

$$\frac{d}{dt}\mathcal{K}(\boldsymbol{\theta},\boldsymbol{\nu})=\boldsymbol{\nu}^*\dot{\boldsymbol{\nu}} = \boldsymbol{\nu}^*[\boldsymbol{\epsilon}-\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu})]=\boldsymbol{\nu}^*\boldsymbol{\epsilon} \tag{7.6}$$

∎

Un-normalized Diagonalized Equations of Motion

An alternative set of diagonalized equations of motion can be obtained by using a slightly different generalized velocity vector defined as

$$\boldsymbol{\xi} = \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{\nu} = [\boldsymbol{I}+\boldsymbol{H}\boldsymbol{\phi}\boldsymbol{K}]^*\dot{\boldsymbol{\theta}} \tag{7.7}$$

The kinetic energy in these coordinates is

$$\mathcal{K}(\boldsymbol{\xi},\dot{\boldsymbol{\xi}})=\frac{1}{2}\dot{\boldsymbol{\xi}}^*\boldsymbol{D}(\boldsymbol{\theta})\dot{\boldsymbol{\xi}} \tag{7.8}$$

The mass matrix now is the block diagonal matrix $\boldsymbol{D}(\boldsymbol{\theta})$. The equations of motion in the new coordinates $(\boldsymbol{\theta},\boldsymbol{\xi})$ are given below.

Lemma 7.3

$$\boldsymbol{D}\dot{\boldsymbol{\xi}} + \boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\xi})= \boldsymbol{\kappa} \tag{7.9}$$

where $\boldsymbol{\kappa} \stackrel{\triangle}{=} \boldsymbol{D}^{\frac{1}{2}}\boldsymbol{\epsilon} = [\boldsymbol{I}-\boldsymbol{H}\boldsymbol{\psi}\boldsymbol{K}]\boldsymbol{T}$ *and*

$$\begin{aligned}\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\xi}) &\stackrel{\triangle}{=} \boldsymbol{D}^{\frac{1}{2}}\left[\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu}) - \boldsymbol{D}^{\frac{1}{2}}\frac{d\boldsymbol{D}^{-\frac{1}{2}}}{dt}\boldsymbol{\nu}\right] \\ &= \boldsymbol{H}\boldsymbol{\psi}\left[\dot{\boldsymbol{\lambda}}\boldsymbol{H}^*\boldsymbol{\xi}-(\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{P}+\tilde{V}^*\boldsymbol{M})V\right] = \boldsymbol{H}\boldsymbol{\psi}\left[\mathring{\boldsymbol{P}}\boldsymbol{H}^*\boldsymbol{\xi}-\tilde{\boldsymbol{\Omega}}_\delta\boldsymbol{P}^+V^+ - \tilde{V}^*\boldsymbol{M}V\right]\end{aligned}$$

∎

The equations of motion in (7.9) are similar to those of the previous section and can be derived readily. They are still diagonal, but they differ from those in (3.1) in two respects. First, although the mass matrix $\boldsymbol{D}$ is diagonal, it is configuration dependent. Moreover, while the Coriolis forces term $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\xi})$ is simpler than $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\nu})$, it is not orthogonal to the generalized velocities vector any more. An $O(\mathcal{N})$ computational algorithm for the components of $\boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\xi})$ is

$$\left\{\begin{array}{rcl} & \dot{\boldsymbol{\lambda}}(0) = 0,\ y(0) = 0 & \\ \textbf{for } \boldsymbol{k} & = & \mathbf{1}\cdots\boldsymbol{\mathcal{N}} \\ X(k) & = & \tilde{\boldsymbol{\Omega}}_\delta(k)\boldsymbol{P}(k) \\ \dot{\boldsymbol{\lambda}}(k) & = & \boldsymbol{\psi}(k,k-1)\dot{\boldsymbol{\lambda}}(k-1)\boldsymbol{\psi}^*(k,k-1) + X(k) + X^*(k) \\ y(k) & = & \boldsymbol{\psi}(k,k-1)y(k-1)+\dot{\boldsymbol{\lambda}}(k)\boldsymbol{H}^*(k)\boldsymbol{\xi}(k)-\left[\tilde{V}^*(k)\boldsymbol{M}(k)+X(k)\right]V(k) \\ \boldsymbol{\mathcal{C}}(k) & = & \boldsymbol{H}(k)y(k) \\ \textbf{end loop} & & \end{array}\right. \tag{7.10}$$

Physical Interpretation of the Coriolis Term

Embedded within the Coriolis force term above is the quantity $\mathring{\boldsymbol{P}}\boldsymbol{H}^*\boldsymbol{\xi} - \tilde{\boldsymbol{\Omega}}_\delta \boldsymbol{P}^+ V^+ - \tilde{V}^*\boldsymbol{M}V$. This quantity consists of three terms, representing three different types of rotation.

The first term $\mathring{\boldsymbol{P}}\boldsymbol{H}^*\boldsymbol{\xi}$ reflects the "broken bicycle chain" effect. This term is dependent only on the past links, which lie outboard of the given link at which the Coriolis force is being computed. Articulation in these outboard joints creates a Coriolis force at the given joint, and the value of this force is dependent on the local time derivative $\mathring{\boldsymbol{P}}$ of the articulated inertia $\boldsymbol{P}$ emerging from the Riccati equation.

The quantity $-\tilde{\boldsymbol{\Omega}}_\delta \boldsymbol{P}^+ V^+$ reflects motion of the base body of the reduced manipulator. This term consequently reflects motion of the future links, in the sense that its value at any given link depends on the motion of all of the inboard links that lie toward the base of the given link.

The quantity $\tilde{\boldsymbol{V}}^*\boldsymbol{M}V$ represents the "present" link. Here, the "present" represents the rotation of the current link k for which the corresponding Coriolis force is being computed. The "present" body k therefore is that link associated with the k^{th} equation in the diagonalized equations of motion.

Thus, the Coriolis force at a given link k is dependent on all of the inertial joint velocities $\boldsymbol{\nu}(i)$ for $i \neq k$. It is a velocity dependent term quite similar to the term $\omega \times \boldsymbol{I}w$ in the dynamics equation for a single rigid body, when generalized to 6 dimensions.

8 Forward Dynamics and Control Applications

$O(\mathcal{N})$ Forward Dynamics

One important application is that of forward dynamics and numerical integration to predict the motion of the manipulator in response to applied moments. An algorithm based upon the unnormalized diagonalized equations of motion in Lemma 7.3 is described here. The acceleration term is:

$$\dot{\boldsymbol{\xi}} \overset{7.9}{=} \boldsymbol{D}^{-1}[\boldsymbol{\kappa} - \boldsymbol{\mathcal{C}}(\boldsymbol{\theta},\boldsymbol{\xi})] \overset{7.1}{=} T - \boldsymbol{H}\boldsymbol{\psi}\zeta \tag{8.1}$$

where

$$\zeta \overset{\triangle}{=} \boldsymbol{K}T + \dot{\boldsymbol{\lambda}}\boldsymbol{H}^*\boldsymbol{\xi} - (\tilde{\boldsymbol{\Omega}}_\delta \boldsymbol{P} + \tilde{V}^*\boldsymbol{M})V \tag{8.2}$$

Algorithm 8.1 *1. Compute the articulated body inertia* $\boldsymbol{P}$ *and the shifted Kalman gain* $\boldsymbol{K}$ *using the Riccati equation in* (4.7)*. Compute also the corresponding time derivative* $\dot{\boldsymbol{\lambda}}$ *using the "sensitivity" equation in Algorithm 6.1.*

2. Compute the time derivatives $\dot{\boldsymbol{\xi}}$ *of the total joint rates* $\boldsymbol{\xi}$ *using the algorithm:*

$$\left\{\begin{array}{lcl} & & \gamma(0) = 0 \\ \textbf{for } \boldsymbol{k} & = & \mathbf{1}\cdots\boldsymbol{\mathcal{N}} \\ \quad \zeta(k) & = & \boldsymbol{K}(k,k-1)T(k-1) + \dot{\boldsymbol{\lambda}}(k)\boldsymbol{H}^*(k)\boldsymbol{\xi}(k) - [\tilde{\boldsymbol{\Omega}}_\delta(k)\boldsymbol{P}(k) + \tilde{V}^*(k)\boldsymbol{M}(k)]V(k) \\ \quad \gamma(k) & = & \boldsymbol{\psi}(k,k-1)\gamma(k-1) + \zeta(k) \\ \quad \dot{\boldsymbol{\xi}}(k) & = & \boldsymbol{D}^{-1}(k)[\boldsymbol{T}(k) - \boldsymbol{H}(k)\gamma(k)] \\ \textbf{end loop} & & \end{array}\right. \tag{8.3}$$

3. *Conduct a numerical integration step to obtain the total joint rates* $\boldsymbol{\xi}$ *at a new time instant.*

4. *Compute the joint-angle rates* $\dot{\boldsymbol{\theta}}$ *and the spatial velocity* $\boldsymbol{V}$ *by the outward recursion in Table 2, modified to account for the fact that* $\boldsymbol{\nu} = D^{-\frac{1}{2}}\boldsymbol{\xi}$.

5. *Integrate the joint-angle rates* $\dot{\boldsymbol{\theta}}$ *to obtain the joint angles* $\boldsymbol{\theta}$ *at the new time instant.*

6. *Go back to the first step and repeat as long as necessary until a prescribed final time has been reached.*

The very first time, V must be computed explicitly (from $\dot{\boldsymbol{\theta}}$ or $\boldsymbol{\xi}$) using one of the algorithms in Table 2. Algorithm 8.1 is similar to those typically [4,18] associated with $O(\mathcal{N})$ forward dynamics. However, it is a significant improvement because it is only a 2-sweep algorithm involving an inward recursion to compute $\dot{\boldsymbol{\xi}}$ followed by an outward recursion to compute $\dot{\boldsymbol{\theta}}$. The Coriolis effects are completely accounted for in the single inward sweep. Previous $O(\mathcal{N})$ algorithms typically [1, 18] require at least 1 or even 2 preliminary inverse dynamics sweeps, prior to utilization of the forward dynamics algorithm.

Decoupled Control

The diagonal equations can also be used to design controllers that are decoupled or non-interacting. The decoupled control approach focuses on the dynamical behavior of the $\boldsymbol{\nu}$ coordinates. Satisfactory performance in the original physical coordinate variables $\dot{\boldsymbol{\theta}}$ follows from this. For example, stability in $\boldsymbol{\nu}$, $\boldsymbol{\epsilon}$ coordinates is equivalent to stability in the original $\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}$ coordinates. The analysis and control design however is simpler because the equations of motion used in the diagonalized design are decoupled. The control problem can be stated in terms of the variables $\boldsymbol{\nu}$ and $\boldsymbol{\epsilon}$ in the diagonalized equations of motion. The problem consists of finding a feedback relationship that determines the input $\boldsymbol{\epsilon}$ in terms of the velocities $\boldsymbol{\nu}$. Once $\boldsymbol{\epsilon}$ is determined, it is possible to go back to physical space to determine the required input moments $\boldsymbol{T}$ by means of the relationship $\boldsymbol{T} = \boldsymbol{m}\boldsymbol{\epsilon}$, and to mechanize this relationship using the inwardly recursive algorithm in Table 3.

Control 8.1 *The rate feedback control*

$$\boldsymbol{\epsilon} = -c\boldsymbol{\nu} \tag{8.4}$$

in which c is a positive diagonal control gain matrix renders the system stable in the sense of Lyapunov.

This result follows by using the kinetic energy as a Lyapunov function and observing that its time derivative (given in Lemma 7.2) can be guaranteed to be negative definite by the choice of the above control approach. This algorithm involves rate feedback only. It can be referred to as a "rate" control algorithm because the feedback quantity is a velocity, in fact, it is a vector of total velocities. It does not guarantee that the manipulator will end up in a prescribed configuration. The following algorithm does this.

Let $\widehat{Y} = \text{col}\left\{\widehat{y}_0, \widehat{y}_1, \widehat{y}_2, \widehat{y}_3\right\}$ be a 12-dimensional vector whose first component $\widehat{y}_0$ is the desired linear position of the end-effector with respect to an inertial reference. The remaining vectors $\widehat{y}_1, \widehat{y}_2, \widehat{y}_3$ are 3 unit vectors which together form an orthonormal basis attached to the end-effector. These three vectors are used to indicate the desired orientation that the end-effector should reach as a result of the control action. Similarly, the end-effector position, in both translation and rotation, is given by $Y(\boldsymbol{\theta}) = \text{col}\left\{y_0(\boldsymbol{\theta}), y_1(\boldsymbol{\theta}), y_2(\boldsymbol{\theta}), y_3(\boldsymbol{\theta})\right\}$, in which the dependence on $\boldsymbol{\theta}$ is shown explicitly. The Jacobian mapping between the joint rates $\dot{\boldsymbol{\theta}}$ and the time derivative of the output Y is

$$\dot{Y} = \boldsymbol{B}^* \boldsymbol{\phi}^* \boldsymbol{H}^* \dot{\boldsymbol{\theta}} \tag{8.5}$$

with B being a suitable linear end-point "pick-off" operator [4], which selects the end-point velocity from the composite spatial velocity vector $\boldsymbol{V} = \boldsymbol{\phi}^* \boldsymbol{H}^* \dot{\boldsymbol{\theta}}$.

The Euclidean norm $\|e\|$ of the error $e \stackrel{\triangle}{=} \widehat{Y} - Y(\boldsymbol{\theta})$ is a measure of the distance between the desired and the actual configuration. The following control algorithm guarantees that the manipulator goes to the prescribed configuration $\widehat{Y}$, while simultaneously driving all the velocities to zero.

Control 8.2 *The feedback control*

$$\boldsymbol{\epsilon} = -c_1 \boldsymbol{\nu} - c_2 \boldsymbol{H} \boldsymbol{\psi} \boldsymbol{B} e \tag{8.6}$$

in which c_1 and c_2 are positive, diagonal control gain matrices, causes the system to reach the prescribed configuration $\widehat{Y}$ and drives the velocities to zero.

This follows easily by taking the time-derivative of the Lyapunov function $\|\boldsymbol{\nu}\|^2 + \|e\|^2$.

The above control approaches require more analysis to include such effects as magnitude bounds on the applied joint moments. The use of diagonal equations of motion for robot control is in its infancy. The main objective of this subsection is to introduce the approach and to provide a few preliminary examples. More comprehensive application of diagonalized models in robot control requires further investigation.

9 Conclusions

The diagonalized equations of motions presented here are very closely related to the body of knowledge [1,3,6,7] recently developed by the authors on spatially recursive algorithms for manipulator dynamics. The present paper complements and builds upon the previous work and explicitly derives the diagonalized Lagrangian equations of motion, which are in addition mechanized by efficient recursive algorithms. The focus here is on the new equations of motion, on the diagonalizing transformations required to obtain them, and on the physical interpretation of the transformed variables. The results presented embed in a single diagonalized equation several of the spatially recursive algorithms previously developed. This is one more step toward increasingly more succinct equations of motion for articulated multibody robotic systems.

Acknowledgments

The research described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration, and has been partly supported by the National Science Foundation Grant ASC 92 19368.

References

[1] G. Rodriguez, "Kalman Filtering, Smoothing and Recursive Robot Arm Forward and Inverse Dynamics," *IEEE Journal of Robotics and Automation*, vol. 3, pp. 624–639, Dec. 1987.

[2] G. Rodriguez, "Random Field Estimation Approach to Robot Dynamics," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, pp. 1081–1093, Sept. 1990.

[3] A. Jain, "Unified Formulation of Dynamics for Serial Rigid Multibody Systems," *Journal of Guidance, Control and Dynamics*, vol. 14, pp. 531–542, May–June 1991.

[4] G. Rodriguez, K. Kreutz-Delgado, and A. Jain, "A Spatial Operator Algebra for Manipulator Modeling and Control," *The International Journal of Robotics Research*, vol. 10, pp. 371–381, Aug. 1991.

[5] G. Rodriguez, A. Jain, and K. Kreutz-Delgado, "Spatial Operator Algebra for Multibody System Dynamics," *Journal of the Astronautical Sciences*, vol. 40, pp. 27–50, Jan.–March 1992.

[6] G. Rodriguez and K. Kreutz-Delgado, "Spatial Operator Factorization and Inversion of the Manipulator Mass Matrix," *IEEE Transactions on Robotics and Automation*, vol. 8, pp. 65–76, Feb. 1992.

[7] A. Jain and G. Rodriguez, "Recursive Flexible Multibody System Dynamics Using Spatial Operators," *Journal of Guidance, Control and Dynamics*, vol. 15, pp. 1453–1466, Nov. 1992.

[8] L. Meirovitch, *Methods of Analytical Dynamics*. McGraw-Hill, New York, 1970.

[9] D. Lovelock and H. Rund, *Tensors, Differential Forms, and Variational Principles*. Dover Publications, New York, 1989.

[10] D. Koditschek, "Robot Kinematics and Coordinate Transformations," in *IEEE Conference on Decision and Control*, pp. 1–4, Dec. 1985.

[11] N. Bedrossian, "Linearizing Coordinate Transformations and Euclidean Systems," in *Workshop on Nonlinear Control of Articulated Flexible Structures*, (Santa Barbara, California), Oct. 1991.

[12] M. Spong, "Remarks on Robot Dynamics: Canonical Transformations and Riemannian Geometry," in *IEEE International Conference on Robotics and Automation*, (Nice, France), pp. 554–559, 1992.

[13] L. Eisenhart, *Riemannian Geometry*. Princeton University Press, Princeton, 1960.

[14] T. Kailath, "The Innovations Approach to Detection and Estimation Theory," *Proceedings of the IEEE*, vol. 58, pp. 680–695, Mar. 1970.

[15] T. Kailath, "A View of Three Decades of Linear Filtering Theory," *IEEE Transactions on Information Theory*, vol. IT–20, pp. 147–181, 1974.

[16] B. D. O. Anderson and J. B. Moore, *Optimal Filtering.* Prentice-Hall Inc., 1979.

[17] J. Luh, M. Walker, and R. Paul, "On-line Computational Scheme for Mechanical Manipulators," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 102, pp. 69–76, June 1980.

[18] R. Featherstone, "The Calculation of Robot Dynamics using Articulated-Body Inertias," *The International Journal of Robotics Research*, vol. 2, pp. 13–30, Spring 1983.

CAPTIONS

Table 1: Physical Interpretation of Spatial Operators

Table 2: $\dot{\theta}$ and ν can be computed recursively from each other

Table 3: ϵ and T can be recursively computed from each other

Table 4: Time Derivatives of Spatial Operators

Figure 1: Illustration of the links and joints in a serial manipulator

Operator	Physical Interpretation
$\boldsymbol{\phi}(k, k-1)$	To-Next-Link Force Transformation
$\boldsymbol{\phi}^*(k, k-1)$	To-Previous-Link Velocity Transformation
$\boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}$	Rigid inward shift force transformation
$\boldsymbol{H}$	Projection to joint axes
$\boldsymbol{H}^*$	Expansion from joint axes
$\boldsymbol{M}$	Rigid link inertia
$\boldsymbol{P}$	Articulated inertia
$\boldsymbol{D} = \boldsymbol{HPH}^*$	Articulated inertia about joint axes
$\boldsymbol{G}$	Kalman gain
$\boldsymbol{K} = \boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}\boldsymbol{G}$	Shifted Kalman gain
$\overline{\boldsymbol{\tau}} = (\boldsymbol{I} - \boldsymbol{GH})$	Joint articulation operator
$\boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}}$	To-next-link articulated shift transformation
$\boldsymbol{\phi} = (\boldsymbol{I} - \boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}})^{-1}$	Rigid manipulator force transformation
$\boldsymbol{\phi}^*$	Rigid manipulator velocity transformation
$\boldsymbol{\psi} = (\boldsymbol{I} - \boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}})^{-1}$	Articulated manipulator force transformation
$\boldsymbol{\psi}^*$	Articulated manipulator velocity transformation

Table 1: Physical Interpretation of Spatial Operators

$\boldsymbol{\nu} = \boldsymbol{m}^*\dot{\boldsymbol{\theta}} = \boldsymbol{D}^{\frac{1}{2}}[\boldsymbol{I} + \boldsymbol{H\phi K}]^*\dot{\boldsymbol{\theta}}$	$\dot{\boldsymbol{\theta}} = \boldsymbol{\ell}^*\boldsymbol{\nu} = [\boldsymbol{I} - \boldsymbol{H\psi K}]^*\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{\nu}$
$V(\mathcal{N}+1) = 0$	$V(\mathcal{N}+1) = 0$
for $\boldsymbol{k} = \boldsymbol{\mathcal{N}n} \cdots \mathbf{1}$	**for** $\boldsymbol{k} = \boldsymbol{\mathcal{N}} \cdots \mathbf{1}$
$V^+(k) = \boldsymbol{\phi}^*(k+1, k)V(k+1)$	$V^+(k) = \boldsymbol{\phi}^*(k+1, k)V(k+1)$
$\boldsymbol{\nu}(k) = \boldsymbol{D}^{\frac{1}{2}}(k)[\dot{\boldsymbol{\theta}}(k) + \boldsymbol{G}^*(k)V^+(k)]$	$\dot{\boldsymbol{\theta}}(k) = \boldsymbol{D}^{-\frac{1}{2}}(k)\boldsymbol{\nu}(k) - \boldsymbol{G}^*(k)V^+(k)$
$V(k) = V^+(k) + \boldsymbol{H}^*(k)\dot{\boldsymbol{\theta}}(k)$	$V(k) = V^+(k) + \boldsymbol{H}^*(k)\dot{\boldsymbol{\theta}}(k)$
end loop	**end loop**

Table 2: $\dot{\boldsymbol{\theta}}$ and $\boldsymbol{\nu}$ can be computed recursively from each other

Figure 1: Illustration of the links and joints in a serial manipulator

$\boldsymbol{\epsilon} = \boldsymbol{\ell T} = \boldsymbol{D}^{-\frac{1}{2}}[\boldsymbol{I} - \boldsymbol{H\psi K}]\boldsymbol{T}$	$\boldsymbol{T} = \boldsymbol{m\epsilon} = [\boldsymbol{I} + \boldsymbol{H\phi K}]\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{\epsilon}$
$z(0) = 0$ **for** $\boldsymbol{k} = \mathbf{1} \cdots \boldsymbol{\mathcal{N}}$ $z(k) = \boldsymbol{\phi}(k, k-1)z^{+}(k-1)$ $\boldsymbol{\epsilon}(k) = \boldsymbol{D}^{-\frac{1}{2}}(k)[\boldsymbol{T}(k) - \boldsymbol{H}(k)z(k)]$ $z^{+}(k) = z(k) + \boldsymbol{G}(k)\boldsymbol{\epsilon}(k)$ **end loop**	$z(0) = 0$ **for** $\boldsymbol{k} = \mathbf{1} \cdots \boldsymbol{\mathcal{N}}$ $z(k) = \boldsymbol{\phi}(k, k-1)z^{+}(k-1)$ $\boldsymbol{T}(k) = \boldsymbol{D}^{\frac{1}{2}}(k)\boldsymbol{\epsilon}(k) + \boldsymbol{H}(k)z(k)$ $z^{+}(k) = z(k) + \boldsymbol{G}(k)\boldsymbol{\epsilon}(k)$ **end loop**

Table 3: $\boldsymbol{\epsilon}$ and $\boldsymbol{T}$ can be recursively computed from each other

	Operator (x)	**Local Derivative** ($\mathring{x}$)	**Inertial Derivative** ($\dot{x}$)
1	$\boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}$	$\boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}\tilde{\boldsymbol{\Omega}}_{\delta}$	$\mathring{\boldsymbol{\mathcal{E}}}_{\boldsymbol{\phi}}+\tilde{\boldsymbol{\Omega}}\boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}} - \boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}\tilde{\boldsymbol{\Omega}}$
2	$\boldsymbol{\phi} = (\boldsymbol{I} - \boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}})^{-1}$	$\boldsymbol{\phi}\mathring{\boldsymbol{\mathcal{E}}}_{\boldsymbol{\phi}}\boldsymbol{\phi} = \boldsymbol{\phi}\boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}\tilde{\boldsymbol{\Omega}}_{\delta}\boldsymbol{\phi}$	$\mathring{\boldsymbol{\phi}}+\boldsymbol{\phi}(\tilde{\boldsymbol{\Omega}}\boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}} - \boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}\tilde{\boldsymbol{\Omega}})\boldsymbol{\phi}$
3	$\boldsymbol{H}$	0	$-\boldsymbol{H}\tilde{\boldsymbol{\Omega}}$
4	$\boldsymbol{M}$	0	$\tilde{\boldsymbol{\Omega}}(k)\boldsymbol{M} - \boldsymbol{M}\tilde{\boldsymbol{\Omega}}$
5	$\boldsymbol{D} = \boldsymbol{HPH}^{*}$	$\boldsymbol{H}\mathring{\boldsymbol{P}}\boldsymbol{H}^{*}$	$\mathring{\boldsymbol{D}}$
6	$\boldsymbol{G}$	$\overline{\boldsymbol{\tau}}\mathring{\boldsymbol{P}}\boldsymbol{H}^{*}\boldsymbol{D}^{-1}$	$\mathring{\boldsymbol{G}} + \tilde{\boldsymbol{\Omega}}\boldsymbol{G}$
7	$\overline{\boldsymbol{\tau}} = \boldsymbol{I} - \boldsymbol{GH}$	$-\mathring{\boldsymbol{G}}\boldsymbol{H}$	$\mathring{\boldsymbol{\tau}}+\tilde{\boldsymbol{\Omega}}\overline{\boldsymbol{\tau}} - \overline{\boldsymbol{\tau}}\tilde{\boldsymbol{\Omega}}$
8	$\boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}} = \boldsymbol{\mathcal{E}}_{\boldsymbol{\phi}}\overline{\boldsymbol{\tau}}$	$\boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}}(-\mathring{\boldsymbol{P}}\boldsymbol{H}^{*}\boldsymbol{D}^{-1}\boldsymbol{H} + \tilde{\boldsymbol{\Omega}}_{\delta}\overline{\boldsymbol{\tau}})$	$\mathring{\boldsymbol{\mathcal{E}}}_{\boldsymbol{\psi}}+\tilde{\boldsymbol{\Omega}}\boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}}-\boldsymbol{\mathcal{E}}_{\boldsymbol{\psi}}\tilde{\boldsymbol{\Omega}}$

Table 4: Time Derivatives of Spatial Operators