Proceedings of the ASME 2009 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2009
August 30-September 2, 2009, San Diego, USA

DETC2009-87663

RECURSIVE ALGORITHMS USING LOCAL CONSTRAINT EMBEDDING FOR MULTIBODY SYSTEM DYNAMICS

Abhinandan Jain
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Pasadena, California 91109
Email: Abhi.Jain@jpl.nasa.gov

ABSTRACT

This paper describes a constraint embedding approach for handling of local closure constraints in multibody system dynamics. The approach uses spatial operator techniques to eliminate local-loop constraints from a system to effectively convert it into a tree-topology system. This conversion to a tree-topology allows the direct application of the host of available techniques including mass matrix factorization and inversion to be applied to the system dynamics. One application is the extension of the well-known recursive $\mathcal{O}(\mathcal{N})$ *forward dynamics for solving the system dynamics of these systems. The algorithms are especially applicable to systems where the constraints are confined to small-subgraphs within the system topology. The paper provides background on the spatial operator approach, the extensions for handling embedded constraints, and concludes with examples of such constraints.*

1 INTRODUCTION

There has been a considerable amount of research over the years on the development of the *Spatial Operator Algebra* methods [1, 2] for the analysis of multibody system dynamics. Key insights that have been developed included analytical techniques for the factorization and inversion of the mass matrix for tree-topology systems. From these results followed the well-known $\mathcal{O}(\mathcal{N})$ forward dynamics algorithms for tree-topology multibody systems [1–3].

Researchers have been able to exploit techniques developed for tree-topology systems to closed-chain topology systems by treating the system as consisting of a tree-topology sub-system together with additional closure constraint. One such example is the forward dynamics algorithm for closed-chain topology systems that consists of the following steps [4]:

1. ignoring the closure constraints and using the $\mathcal{O}(\mathcal{N})$ algorithm to solve for the "free" unconstrained accelerations for the tree-topology sub-system;
2. using the tree-topology solution to compute a correction force that enforces the closure-constraints;
3. correcting the unconstrained accelerations with correction accelerations resulting from the correction forces.

The correction step (2) requires the computation of the reflected inertias at the closure nodes, referred to as the operational space inertia in order to obtain the correction forces.

The above algorithm for closed-chain topologies has some significant drawbacks when compared with the $\mathcal{O}(\mathcal{N})$ algorithms for tree-topology systems. These include the use of a non-minimal set of generalized coordinates and velocities, the need for a differential-algebraic solver and/or constraint stabilization technique for propagating the equations of motion, and the multi-pass algorithm described above. Another major consequence from the use of a non-minimal set of generalized velocities that the notion of a mass-matrix is no longer directly applicable to closed-chain systems, and thus we are unable to take much advantage of the spatial operator mass-matrix factorization and inversion techniques any more. Our goal in this paper is to reformulate the equations of motion for closed-chain systems to overcome these limitations and recast them into a form that recovers all the benefits of a tree-topology formulation. This recasting extends the direct applicability of the large range spatial operator results, algorithms and techniques developed for tree-topology systems to systems with closed-chain topology. This research is inspired by the *Recursive Coordinate Reduction (RCR)* algorithm [5] that described a way to directly absorb closure constraints into the $O(\mathcal{N})$ forward dynamics algorithms bu adding additional phantom-bodies to the system.

The reformulation described in this paper is most beneficial for the class of multibody systems where the closure-constraints are local, i.e., where the the constraints are confined to small, connected sub-groups of bodies within the system. Important examples of such local closure-constraints are four-bar linkages, geared motors, differential suspensions etc. elements within a multibody system. This paper shows how one can embed these closure-constraints directly into the system dynamics and effectively replace the sub-group of bodies with virtual aggregate bodies. This step transforms the original closed-chain topology system into a tree-topology one with the closure body subgroups replaced by aggregated bodies. The transformation to tree-topology systems allows us to apply all available techniques for tree-topology systems to the new system's dynamics including the extension of the tree-topology $O(\mathcal{N})$ forward dynamics algorithm. The structural impact on the class of recursive dynamics algorithms is minimal in that the changes are limited to the steps encountered when crossing an aggregated body in the system. While the results here are quite general, we believe that this approach is especially effective for "local" loops since the aggregation step is typically simple for these cases.

We begin with a brief overview of the spatial operator algebra approach to setting up the equations of motion for a multibody system, followed by the steps leading to the derivation of $O(\mathcal{N})$ forward-dynamics algorithms for tree-topology systems. We then move on to the case of systems with local closure constraints and describe the embedding technique. We derive modifications for the $O(\mathcal{N})$ algorithm needed to handle the embedded constraints. We conclude with specific examples of embedding local closure-constraints.

2 OVERVIEW OF SPATIAL OPERATORS FOR SERIAL CHAIN SYSTEMS

The aim of this section is to briefly summarize the essential ideas underlying spatial operators leading up to the Newton-Euler Operator Factorization $\mathcal{M}(\theta) = H\phi\mathbf{M}\phi^*H^*$ of the manipulator mass matrix. While this is done here for a serial chain manipulator, the factorization results apply to more general class of complex joint-connected mechanical systems, including tree configurations with flexible links and joints [6].

Consider a serial manipulator with $\mathcal{N}$ rigid links. The links are numbered in increasing order from tip to base. The outermost 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 $\mathbb{O}_k$ is attached to the inboard joint, and another frame $\mathbb{O}^+_{k-1}$ is attached to the outboard joint. Frame $\mathbb{O}_k$ 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 $\mathbb{O}_k$ with respect to frame $\mathbb{O}^+_k$. 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 $\theta(k)$. Extension to joints with more rotational and translational degrees-of-freedom is straightforward [4].

The transformation operator $\phi(k,k-1)$ between the $\mathbb{O}_{k-1}$ and $\mathbb{O}_k$ frames is

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

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

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

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

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

The recursive Newton–Euler equations of motion are [2, 7]

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

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

where $\mathcal{T}(k)$ is the applied moment at joint k. The nonlinear, velocity dependent terms $\mathfrak{a}(k)$ and $\mathfrak{b}(k)$ are respectively the Coriolis acceleration and the gyroscopic force terms for the k^{th} link.

The "stacked" notation $\theta = \mathrm{col}\left\{\theta(k)\right\} \in \mathcal{R}^{\mathcal{N}}$ is used to simplify the above recursive Newton-Euler equations. This notation [8] eliminates the arguments k associated with the individual links by defining composite vectors, such as θ, which apply to the entire manipulator system. We define

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

In this notation, the equations of motion are [2, 9]:

$$V = \phi^* H^* \dot{\theta}; \qquad \alpha = \phi^*[H^*\ddot{\theta} + \mathfrak{a}] \tag{2}$$

$$f = \phi[\mathbf{M}\alpha + \mathfrak{b}]; \qquad \mathcal{T} = Hf = \mathcal{M}\ddot{\theta} + \mathcal{C} \tag{3}$$

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

$$\phi = (\mathbf{I} - \mathcal{E}_\phi)^{-1} = \begin{pmatrix} \mathbf{I} & 0 & \dots 0 \\ \phi(2,1) & \mathbf{I} & \dots 0 \\ \vdots & \vdots & \ddots \vdots \\ \phi(n,1) & \phi(n,2) & \dots \mathbf{I} \end{pmatrix} \tag{4}$$

with $\phi(i,j) = \phi(i,i-1)\cdots\phi(j+1,j)$ for $i > j$. The shift operator $\mathcal{E}_\phi \in \mathcal{R}^{6\mathcal{N}\times 6\mathcal{N}}$ is defined as

$$\mathcal{E}_\phi = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ \phi(2,1) & 0 & \dots & 0 & 0 \\ 0 & \phi(3,2) & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \phi(\mathcal{N},\mathcal{N}-1) & 0 \end{pmatrix} \tag{5}$$

Using spatial operators one can obtain operator factorizations of the mass matrix and its inverse as follows:

$$\begin{aligned} \mathcal{M} &= H\phi\mathbf{M}\phi^*H^* \\ &= [\mathbf{I} + H\phi\mathcal{K}]\mathcal{D}[\mathbf{I} + H\phi\mathcal{K}]^* \\ [\mathbf{I} + H\phi\mathcal{K}]^{-1} &= \mathbf{I} - H\psi\mathcal{K} \\ \mathcal{M}^{-1} &= [\mathbf{I} - H\psi\mathcal{K}]^*\mathcal{D}^{-1}[\mathbf{I} - H\psi\mathcal{K}] \end{aligned} \tag{6}$$

These identifies have been used extensively [1, 2, 8–13], to develop a variety of spatially recursive algorithms for forward dynamics, for both rigid and flexible multi-body systems of arbitrarily specified topologies, as well as closed-form analytical expressions for the inverse of the mass matrix. The spatial operators ψ, $\mathcal{D}$ correspond to a suitably defined spatially recursive Kalman filter, with the spatial operator $\mathcal{K}$ representing the Kalman gain for this filter. We also refer to these operators ψ, $\mathcal{D}$ and $\mathcal{K}$ as"articulated" quantities, because of their relationship to the articulated inertias first introduced by [3].

The mass matrix factor $[\mathbf{I} + H\phi\mathcal{K}]$ is a square, invertible matrix and so is its inverse $[\mathbf{I} - H\psi\mathcal{K}]$.

3 $O(\mathcal{N})$ FORWARD DYNAMICS

Using the expression for the mass matrix inverse in Eq. (6), and some additional spatial operator identities, it has been shown that [1]

$$\ddot{\theta} = \mathcal{M}^{-1}(\mathcal{T} - \mathcal{C}) = [\mathbf{I} - H\psi\mathcal{K}]^*\mathcal{D}^{-1}[\mathbf{I} - H\psi\mathcal{K}](\mathcal{T} - \mathcal{C}) \tag{7}$$

This expression can be broken into the following sequence of intermediate quantities:

$$
\begin{aligned}
&\mathfrak{z} = \psi(\mathcal{K}\mathcal{T}+\mathcal{P}\mathfrak{a}+\mathfrak{b}) \\
&\epsilon = \quad \mathcal{T}-H\mathfrak{z} \quad = \mathcal{T}-H\psi(\mathcal{K}\mathcal{T}+\mathcal{P}\mathfrak{a}+\mathfrak{b}) \\
&\nu = \quad \mathcal{D}^{-1}\epsilon \quad = \mathcal{D}^{-1}[\mathcal{T}-H\psi(\mathcal{K}\mathcal{T}+\mathcal{P}\mathfrak{a}+\mathfrak{b})] \\
&\alpha = \quad \psi(H^*\nu+\mathfrak{a}) \quad = \psi^*(H^*\mathcal{D}^{-1}[\mathcal{T}-H\psi(\mathcal{K}\mathcal{T}+\mathcal{P}\mathfrak{a}+\mathfrak{b})]+\mathfrak{a}) \\
&\ddot{\theta} = \quad \nu-\mathcal{K}^*\alpha \quad = [\mathbf{I}-H\psi\mathcal{K}]^*\mathcal{D}^{-1}[\mathcal{T}-H\psi(\mathcal{K}\mathcal{T} \\
&\qquad\qquad +\mathcal{P}\mathfrak{a}+\mathfrak{b})]-\mathcal{K}^*\psi^*\mathfrak{a}
\end{aligned}
\tag{8}
$$

These operator expressions can be converted into recursive computational algorithms without requiring the explicit computation of the component operators. The resulting $O(\mathcal{N})$ forward dynamics procedure is as follows:

$$
\begin{cases}
\qquad \mathcal{P}(0)=0, \quad \mathfrak{z}(0)=0, \quad \mathcal{T}(0)=0, \quad \overline{\tau}(0)=\mathbf{0} \\
\textbf{for } k = 1\cdots\mathcal{N} \\
\quad \psi(k,k-1)=\phi(k,k-1)\overline{\tau}(k-1) \\
\quad \mathfrak{z}(k)=\psi(k,k-1)\mathfrak{z}(k-1)+\mathcal{P}(k)\mathfrak{a}(k)+\mathfrak{b}(k) \\
\qquad\quad +\mathcal{K}(k,k-1)T(k-1) \\
\quad \mathcal{P}(k)=\psi(k,k-1)\mathcal{P}(k-1)\psi^*(k,k-1)+M(k) \\
\quad \mathcal{D}(k)=H(k)\mathcal{P}(k)H^*(k) \\
\quad \mathcal{G}(k)=\mathcal{P}(k)H^*(k)\mathcal{D}^{-1}(k) \\
\quad \mathcal{K}(k+1,k)=\phi(k+1,k)\mathcal{G}(k) \\
\quad \overline{\tau}(k)=\mathbf{I}-\mathcal{G}(k)H(k) \\
\quad \epsilon(k)=\check{\mathcal{T}}(k)-H(k)\mathfrak{z}(k) \\
\quad \nu(k)=\mathcal{D}^{-1}(k)\epsilon(k) \\
\textbf{end loop}
\end{cases}
\tag{9}
$$

$$
\begin{cases}
\qquad \alpha(n+1)=\mathbf{0} \\
\textbf{for } k = \mathcal{N}\cdots 1 \\
\quad \ddot{\theta}(k)=\nu(k)-\mathcal{K}^*(k+1,k)\alpha(k+1) \\
\quad \alpha(k)=\psi^*(k+1,k)\alpha(k+1)+H^*(k)\nu(k)+\mathfrak{a}(k) \\
\textbf{end loop}
\end{cases}
$$

This algorithm includes the recursive steps for the computation of the $\mathcal{P}(.)$'s and $\mathfrak{z}(.)$. The fact that the computational cost of this algorithm is $O(\mathcal{N})$ follows from the fact that the computational cost of each of the steps in the above algorithm is of fixed size, and each of these steps is carried out $\mathcal{N}$ times during the course of the algorithm.

As we see here, this forward dynamics algorithm does not require the explicit computation of either $\mathcal{M}$ or $\mathcal{C}$. Indeed it did not require the explicit computation of any of the spatial operators either. It illustrates the ease with which the high level operator level manipulations can be used to establish key identities and results, and at a later stage when the time for computations arises, these results can be mapped into highly efficient computational algorithms.

4 LOCAL CONSTRAINT EMBEDDING

4.1 Constraint Sub-Groups

Let us assume that within the multibody system, we have a a sub-group of bodies with some closure constraints among them. One consequence of the constraints is that the effective degrees of freedom associated with the bodies in this sub-group is less than the sum total of the hinge degrees of freedom . Examples of such local loops include constraints associated with geared motors, 4-bar linkages/wishbone suspensions, differentials etc. The presence of the local loops implies that the system is no longer a tree-topology system.

Our first step is to isolate this sub-group in order to clearly define its internal kinematics/dynamics relationships, as well as its coupling to the rest of the system. For the purpose of exposition, we assume that the links have been numbered so that the indices for the links in the sub-group range from from i to j with $i > j$. Assume that the i^{th} link in the sub-group is the child of the $(i+1)^{th}$ link and the $(j-1)^{th}$ link is the child of the j^{th} link in the sub-group.

Define the stacked spatial velocities vector $\mathcal{V}_{\mathfrak{G}} = \text{col}\left\{\mathcal{V}(i),\cdots\mathcal{V}(j)\right\}$, and the stacked generalized velocities vector $\dot{\theta}_{\mathfrak{G}} = \text{col}\left\{\dot{\theta}(i),\cdots\dot{\theta}(j)\right\}$. The $\mathcal{V}_{\mathfrak{G}}$ and $\dot{\theta}_{\mathfrak{G}}$ vectors are subvectors of the full $\mathcal{V}$ and $\dot{\theta}$ vectors corresponding to only the links in the sub-group. Then we have,

$$
\begin{aligned}
\mathcal{V}_{\mathfrak{G}} &= \mathcal{E}^*_{\phi_{\mathfrak{G}}}\mathcal{V}_{\mathfrak{G}} + E_{\mathfrak{G}}{}^*\mathcal{V}(i+1) + H^*{}_{\mathfrak{G}}\dot{\theta}_{\mathfrak{G}} \\
\mathcal{V}(j-1) &= \phi^*(j,j-1)\mathcal{B}_{\mathfrak{G}}{}^*\mathcal{V}_{\mathfrak{G}} + H^*(j-1)\dot{\theta}(j-1)
\end{aligned}
\tag{10}
$$

Here $\mathcal{E}^*_{\phi_{\mathfrak{G}}}$ is the block element of $\mathcal{E}_{\phi}$ corresponding to just the bodies in the sub-group. $E_{\mathfrak{G}}$ and $\mathcal{B}_{\mathfrak{G}}$ are also sub-blocks of $\mathcal{E}_{\phi}$ that denote the coupling of the sub-group to links $i+1$ and $j-1$ respectively. In essence, Eq. (10) is a block-partitioned restatement of the system level velocity relationships. The first equation defines how the parent body's velocity couples into the sub-group while the second one defines how the sub-groups velocities couple into the outboard bodies. Note that the loop constraint imposes internal consistency conditions on the elements of $\mathcal{V}_{\mathfrak{G}}$, and these conditions are met by admissible $\dot{\theta}_{\mathfrak{G}}$ generalized velocities that are consistent with the constraints. Due to the internal-dependency, there are multiple (and equally valid) options for defining $\mathcal{E}^*_{\phi_{\mathfrak{G}}}$.

Continuing on, we now define sub-group rigid body transformation operator $\phi_{\mathfrak{S}}$ analogous to Eq. (4) as

$$\phi_{\mathfrak{S}} \stackrel{\triangle}{=} (\mathbf{I} - \mathcal{E}_{\phi_{\mathfrak{S}}})^{-1} \tag{11}$$

With this we can rewrite the first equation in Eq. (10) as

$$\mathcal{V}_{\mathfrak{S}} \stackrel{10}{=} \phi^*_{\mathfrak{S}} E_{\mathfrak{S}}{}^* \mathcal{V}(i+1) + \phi^*_{\mathfrak{S}} H^*{}_{\mathfrak{S}} \dot{\theta}_{\mathfrak{S}} \tag{12}$$

Note that $\phi_{\mathfrak{S}}$ is a sub-block of the full ϕ for the sub-graph links. In effect, we are partitioning all the spatial operators to create single block entries to represent the sub-group. Indeed, we can now think of the sub-group as an **aggregate link** with associated link transformation matrix $\phi^*_{\mathfrak{S}} E_{\mathfrak{S}}{}^*$ and joint map matrix $\phi^*_{\mathfrak{S}} H^*{}_{\mathfrak{S}}$. The aggregate link for the sub-graph then has spatial velocity defined by $\mathcal{V}_{\mathfrak{S}}$ and its associated generalized velocity coordinates are $\dot{\theta}_{\mathfrak{S}}$. Assuming that the topology of the system external to the sub-group is a tree, the new system with the aggregate body has been transformed into a form resembling a tree-topology structure.

4.2 Embedding the sub-group constraints

Having seen how to decompose and isolate sub-groups of bodies as aggregate bodies we now turn to the subject of handling closure-constraints within these sub-groups. We describe here the process of embedding these constraints directly into the dynamics model so that we can complete the transformation to the simpler tree topology model. However this proxy link is no longer a regular link, but instead a **aggregate link** responsible for appropriately handling the contribution of the sub-group of links in the system dynamics. Note, that the new model is an exact replacement for the original model and no approximations are involved.

Due to the internal constraints within the sub-group, not all elements of $\dot{\theta}_{\mathfrak{S}}$ are independent. Hence, there exists a (configuration dependent) mapping $X_{\mathfrak{S}}$ such that

$$\begin{aligned} \dot{\theta}_{\mathfrak{S}} = X_{\mathfrak{S}} \dot{\theta}_{R\mathfrak{S}} \implies H^*{}_{\mathfrak{S}} \dot{\theta}_{\mathfrak{S}} = H^*{}_{R\mathfrak{S}} \dot{\theta}_{R\mathfrak{S}} \\ \text{where} \quad H^*{}_{R\mathfrak{S}} \stackrel{\triangle}{=} H^*{}_{\mathfrak{S}} X_{\mathfrak{S}} \end{aligned} \tag{13}$$

In the above, $\dot{\theta}_{R\mathfrak{S}}$ denotes the truly independent generalized velocity sub-vector of $\dot{\theta}_{\mathfrak{S}}$. We will later explore how one might go about computing $X_{\mathfrak{S}}$ for sub-graphs.

Using Eq. (13) in Eq. (12) we have the following new velocity transformation relationship for the sub-group in terms of independent generalized velocities:

$$\begin{aligned} \mathcal{V}_{\mathfrak{S}} &\stackrel{11,12,13}{=} \phi^*_{\mathfrak{S}} E_{\mathfrak{S}}{}^* \mathcal{V}(i+1) + \phi^*_{\mathfrak{S}} H^*{}_{R\mathfrak{S}} \dot{\theta}_{R\mathfrak{S}} \\ &= \phi^*_{\mathfrak{S}} E_{\mathfrak{S}}{}^* \mathcal{V}(i+1) + \underline{H}^*_{R\mathfrak{S}} \dot{\theta}_{R\mathfrak{S}} \end{aligned} \tag{14}$$

where $\underline{H}^*_{R\mathfrak{S}}$ is defined as

$$\underline{H}^*_{R\mathfrak{S}} \stackrel{\triangle}{=} \phi^*_{\mathfrak{S}} H^*{}_{R\mathfrak{S}} \tag{15}$$

With this new-relationship, we can re-express all of the system-level spatial operators into new versions where the blocks corresponding to aggregate body links are treated as coming from a single link.

Analogous expressions at the acceleration level are described below. First note that

$$\ddot{\theta}_{\mathfrak{S}} = X_{\mathfrak{S}} \ddot{\theta}_{R\mathfrak{S}} + \dot{X}_{\mathfrak{S}} \dot{\theta}_{R\mathfrak{S}} \tag{16}$$

Hence the acceleration level expressions take the form:

$$\begin{aligned} \alpha_{\mathfrak{S}} &= \mathcal{E}^*_{\phi_{\mathfrak{S}}} \alpha_{\mathfrak{S}} + E_{\mathfrak{S}}{}^* \alpha(i+1) + H^*{}_{\mathfrak{S}} \ddot{\theta}_{\mathfrak{S}} + \mathfrak{a}_{\mathfrak{S}} \\ \implies \alpha_{\mathfrak{S}} &= \phi^*_{\mathfrak{S}} E_{\mathfrak{S}}{}^* \alpha(i+1) + \phi^*_{\mathfrak{S}} H^*{}_{\mathfrak{S}} \ddot{\theta}_{\mathfrak{S}} + \phi^*_{\mathfrak{S}} \mathfrak{a}_{\mathfrak{S}} \\ &= \phi^*_{\mathfrak{S}} E_{\mathfrak{S}}{}^* \alpha(i+1) + \underline{H}^*_{R\mathfrak{S}} \ddot{\theta}_{R\mathfrak{S}} + \phi^*_{\mathfrak{S}} \mathfrak{a}_{\mathfrak{S}} \\ &\quad + \phi^*_{\mathfrak{S}} H^*{}_{\mathfrak{S}} \dot{X}_{\mathfrak{S}} \dot{\theta}_{R\mathfrak{S}} \\ &= \phi^*_{\mathfrak{S}} E_{\mathfrak{S}}{}^* \alpha(i+1) + \underline{H}^*_{R\mathfrak{S}} \ddot{\theta}_{R\mathfrak{S}} + \mathfrak{a}'_{\mathfrak{S}} \end{aligned} \tag{17}$$

where

$$\mathfrak{a}'_{\mathfrak{S}} \stackrel{\triangle}{=} \phi^*_{\mathfrak{S}} (\mathfrak{a}_{\mathfrak{S}} + H^*{}_{\mathfrak{S}} \dot{X}_{\mathfrak{S}} \dot{\theta}_{R\mathfrak{S}}) \tag{18}$$

Note that the structure of Eq. (14) and Eq. (17) resemble those of the velocity and acceleration relations in Eq. (1) for tree-topology systems. We can continue on to transform the spatial force expressions as well. For this we will need the spatial inertia term, $\mathbf{M}$, for the aggregate body. $\mathbf{M}$ is simply the sub-block of $M_{\mathfrak{S}}$ corresponding to the bodies sub-group for the aggregate body. We skip the details since the development is straightforward.

5 FORWARD DYNAMICS WITH CONSTRAINT EMBEDDING

With these changes, we once again have a new reduced, mass matrix for the system corresponding to just the independent generalized velocities and forces. The important point to note is that a consequence of the embedding has been that the system and the mass matrix continue to have the familiar structure for tree-topology systems. As a consequence, we can repeat the steps leading to the Innovations Factorization and inversion of the mass matrix for the reduced mass matrix to obtain the corresponding operator factors and inverse operator expressions for the reduced mass matrix. Note however, that the a consequence of the embedding has been that some of the blocks in the new operators are much larger than normal since they encompass terms for the full sub-group. The large the sub-groups, the larger are these block-elements.

With this setup, all of the inverse and forward dynamics results continue to apply for the new operators defined by embedding the constraints. We now look in more detail at the $O(\mathcal{N})$ forward dynamics algorithm in Eq. (9) to our new reformulated system with aggregate links. The articulated body inertia recursive forward dynamics algorithm for the new embedded system has the form:

$$\begin{aligned}
\mathcal{P}^+(j-1) &= \overline{\tau}(j-1)\mathcal{P}(j-1) \\
\mathcal{P}_{\mathfrak{G}} &= \mathcal{B}_{\mathfrak{G}}\phi(j,j-1)\mathcal{P}^+(j-1)\phi^*(j,j-1)\mathcal{B}_{\mathfrak{G}}{}^* + M_{\mathfrak{G}} \\
\mathcal{D}_{\mathfrak{G}} &= \underline{H}_{R\mathfrak{G}}\mathcal{P}_{\mathfrak{G}}\underline{H}^*_{R\mathfrak{G}} \\
\mathcal{G}_{\mathfrak{G}} &= \mathcal{P}_{\mathfrak{G}}\underline{H}^*_{R\mathfrak{G}}\mathcal{D}^{-1}_{\mathfrak{G}} \\
\tau_{\mathfrak{G}} &= \mathcal{G}_{\mathfrak{G}}\underline{H}_{R\mathfrak{G}} \\
\mathcal{P}^+_{\mathfrak{G}} &= \mathcal{P}_{\mathfrak{G}} - \tau_{\mathfrak{G}}\mathcal{P}_{\mathfrak{G}} \\
\mathcal{P}(i+1) &= E_{\mathfrak{G}}\phi_{\mathfrak{G}}\mathcal{P}^+_{\mathfrak{G}}\phi^*_{\mathfrak{G}}E_{\mathfrak{G}}{}^* + M(i+1)
\end{aligned} \tag{19}$$

Note that $\mathcal{D}_{\mathfrak{G}}$ can be re-expressed as

$$\begin{aligned}
\mathcal{D}_{\mathfrak{G}} = X_{\mathfrak{G}}{}^*(H_{\mathfrak{G}}\phi_{\mathfrak{G}}\mathcal{P}_{\mathfrak{G}}\phi^*_{\mathfrak{G}}H^*{}_{\mathfrak{G}})X_{\mathfrak{G}} = X_{\mathfrak{G}}{}^*\mathcal{M}_{\mathfrak{G}}X_{\mathfrak{G}} \\
\text{where} \quad \mathcal{M}_{\mathfrak{G}} \stackrel{\triangle}{=} H_{\mathfrak{G}}\phi_{\mathfrak{G}}\mathcal{P}_{\mathfrak{G}}\phi^*_{\mathfrak{G}}H^*{}_{\mathfrak{G}}
\end{aligned} \tag{20}$$

The inner term $\mathcal{M}_{\mathfrak{G}}$ has the structure of the mass matrix of the sub-group's tree. The one major difference from the true subtree's mass matrix is that the central body spatial inertia operator is $\mathcal{P}_{\mathfrak{G}}$ instead of the normal $M_{\mathfrak{G}}$ term. However, the structural properties of a tree-topology continue to hold as do results such as the composite rigid body decomposition, operator-based mass matrix factorization and inversion etc. $\mathcal{D}_{\mathfrak{G}}$ is the reduced mass matrix for the sub-tree projected down down to the independent degrees of freedom for the sub-tree. In this sub-tree, link $(i+1)$ is serves as the "root" inertial frame, and the sub-graph bodies directly attached to this link are independent base-bodies attached to the inertial frame. Due to the implicit cuts, the sub-group multi-body system has a tree-topology structure. Also, note that the mass of the link j includes the articulated body contribution from the sub-tree rooted at link $j-1$.

The accompanying vector recursions for the articulated body inertia algorithm in Eq. (9) take the following form for the embedded constraints. First the step from body $(j-1)$ to the i^{th} aggregate body:

$$\begin{aligned}
\mathfrak{z}^+(i-1) &= \mathfrak{z}(i-1) + \mathcal{G}(i-1)\epsilon(i-1) \\
\mathfrak{z}_{\mathfrak{G}} &= \mathcal{B}_{\mathfrak{G}}\phi(j,j-1)\mathfrak{z}^+(i-1) + \mathfrak{b}_{\mathfrak{G}} + \mathcal{P}_{\mathfrak{G}}\mathfrak{a}'_{\mathfrak{G}} \\
\epsilon_{\mathfrak{G}} &= \mathcal{T}(i) - \underline{H}_{R\mathfrak{G}}\mathfrak{z}_{\mathfrak{G}} \\
\nu_{\mathfrak{G}} &= \mathcal{D}^{-1}_{\mathfrak{G}}\epsilon_{\mathfrak{G}}
\end{aligned} \tag{21}$$

The recursion step from the i^{th} body to body $(j+1)$ is as follows:

$$\begin{aligned}
\mathfrak{z}^+_{\mathfrak{G}} &= \mathfrak{z}_{\mathfrak{G}} + \mathcal{G}_{\mathfrak{G}}\epsilon_{\mathfrak{G}} \\
\mathfrak{z}(i+1) &= E_{\mathfrak{G}}\phi_{\mathfrak{G}}\mathfrak{z}^+_{\mathfrak{G}} + \mathfrak{b}(i+1) + \mathcal{P}(i+1)\mathfrak{a}(i+1) \\
\epsilon(i+1) &= \mathcal{T}(i+1) - H(i+1)\mathfrak{z}(i+1) \\
\nu(i+1) &= \mathcal{D}^{-1}(i+1)\epsilon(i+1)
\end{aligned} \tag{22}$$

The base-to-tip accelerations sweeps steps also are altered as follows. First the steps from body $(i+1)$ to the i^{th} aggregate body:

$$\begin{aligned}
\alpha^+{}_{\mathfrak{G}} &= \phi^*_{\mathfrak{G}}E_{\mathfrak{G}}{}^*\alpha(i+1) \\
\ddot{\theta}_{R\mathfrak{G}} &= \nu_{\mathfrak{G}} - \mathcal{G}_{\mathfrak{G}}{}^*\alpha^+{}_{\mathfrak{G}} \\
\alpha_{\mathfrak{G}} &= \alpha^+{}_{\mathfrak{G}} + \underline{H}^*_{R\mathfrak{G}}\ddot{\theta}_{R\mathfrak{G}} + \mathfrak{a}'_{\mathfrak{G}}
\end{aligned} \tag{23}$$

The step from the i^{th} aggregate body to body $(j-1)$ is as follows:

$$\begin{aligned}
\alpha^+(i-1) &= \mathcal{B}_{\mathfrak{G}}{}^*\alpha_{\mathfrak{G}} \\
\ddot{\theta}(i-1) &= \nu(i-1) - \mathcal{G}^*(i-1)\alpha^+(i-1) \\
\alpha(i-1) &= \alpha^+(i-1) + H^*(i-1)\ddot{\theta}(i-1) + \mathfrak{a}(i-1)
\end{aligned} \tag{24}$$

5.1 Alternative form of forward dynamics steps

Defining

$$\underline{\mathcal{P}}_{\mathfrak{G}} \stackrel{\triangle}{=} \phi_{\mathfrak{G}}\mathcal{P}_{\mathfrak{G}}\phi^*_{\mathfrak{G}}, \tag{25}$$

we can write a slightly rearranged version of Eq. (19) as

$$\begin{aligned}
\mathcal{P}^+(j-1) &= \overline{\tau}(j-1)\mathcal{P}(j-1) \\
\underline{\mathcal{P}}_{\mathfrak{S}} &= \phi_{\mathfrak{S}}\left[\mathcal{B}_{\mathfrak{S}}\phi(j,j-1)\mathcal{P}^+(j-1)\phi^*(j,j-1)\mathcal{B}_{\mathfrak{S}}{}^* + M_{\mathfrak{S}}\right]\phi^*_{\mathfrak{S}} \\
&\quad (= \phi_{\mathfrak{S}}\mathcal{P}_{\mathfrak{S}}\phi^*_{\mathfrak{S}}) \\
\mathcal{D}_{\mathfrak{S}} &= H_{R\mathfrak{S}}\underline{\mathcal{P}}_{\mathfrak{S}}H^*{}_{R\mathfrak{S}} \\
\underline{\mathcal{G}}_{\mathfrak{S}} &= \underline{\mathcal{P}}_{\mathfrak{S}}H^*{}_{R\mathfrak{S}}\mathcal{D}^{-1}_{\mathfrak{S}} \quad (= \phi_{\mathfrak{S}}\mathcal{G}_{\mathfrak{S}}) \\
\underline{\tau}_{\mathfrak{S}} &= \underline{\mathcal{G}}_{\mathfrak{S}}H_{R\mathfrak{S}} \quad (= \phi_{\mathfrak{S}}\tau_{\mathfrak{S}}\phi_{\mathfrak{S}}{}^{-1}) \\
\underline{\mathcal{P}^+}_{\mathfrak{S}} &= \underline{\mathcal{P}}_{\mathfrak{S}} - \underline{\tau}_{\mathfrak{S}}\underline{\mathcal{P}}_{\mathfrak{S}} \quad (= \phi_{\mathfrak{S}}\mathcal{P}^+_{\mathfrak{S}}\phi^*_{\mathfrak{S}}) \\
\mathcal{P}(i+1) &= E_{\mathfrak{S}}\underline{\mathcal{P}^+}_{\mathfrak{S}}E_{\mathfrak{S}}{}^* + M(i+1)
\end{aligned} \tag{26}$$

The above corresponds to the following alternative definition $\underline{\mathcal{V}}_{\mathfrak{S}}$ of the sub-graph's spatial velocity:

$$\mathcal{V}_{\mathfrak{S}} = \phi^*_{\mathfrak{S}}\underline{\mathcal{V}}_{\mathfrak{S}} \quad \text{or} \quad \underline{\mathcal{V}}_{\mathfrak{S}} \overset{11}{=} (\mathbf{I} - \mathcal{E}^*_{\phi_{\mathfrak{S}}})\mathcal{V}_{\mathfrak{S}} \tag{27}$$

Form Eq. (12) it follows then that the velocity equations have the form

$$\begin{aligned}
\underline{\mathcal{V}}_{\mathfrak{S}} &= E_{\mathfrak{S}}{}^*\mathcal{V}(i+1) + H^*{}_{R\mathfrak{S}}\dot{\theta}_{R\mathfrak{S}} \\
\mathcal{V}(j-1) &= \phi^*(j,j-1)\mathcal{B}_{\mathfrak{S}}{}^*\phi^*_{\mathfrak{S}}\underline{\mathcal{V}}_{\mathfrak{S}} + H^*(j-1)\dot{\theta}(j-1)
\end{aligned} \tag{28}$$

while the corresponding acceleration equations have the form

$$\begin{aligned}
\underline{\alpha}_{\mathfrak{S}} &= E_{\mathfrak{S}}{}^*\alpha(i+1) + H^*{}_{\mathfrak{S}}\ddot{\theta}_{\mathfrak{S}} + \underline{\mathfrak{a}}_{\mathfrak{S}} + H^*{}_{\mathfrak{S}}\dot{X}_{\mathfrak{S}}\dot{\theta}_{R\mathfrak{S}} \\
&= E_{\mathfrak{S}}{}^*\alpha(i+1) + H^*{}_{R\mathfrak{S}}\ddot{\theta}_{R\mathfrak{S}} + \underline{\mathfrak{a}'}_{\mathfrak{S}}
\end{aligned} \tag{29}$$

Also define

$$\underline{\mathfrak{z}}_{\mathfrak{S}} \overset{\triangle}{=} \phi_{\mathfrak{S}}\mathfrak{z}_{\mathfrak{S}} \quad \text{and} \quad \underline{\mathfrak{z}}^+_{\mathfrak{S}} \overset{\triangle}{=} \phi_{\mathfrak{S}}\mathfrak{z}^+_{\mathfrak{S}} \tag{30}$$

Then from Eq. (21) we have

$$\begin{aligned}
\underline{\mathfrak{z}}_{\mathfrak{S}} &= \phi_{\mathfrak{S}}\left[\mathcal{B}_{\mathfrak{S}}\phi(j,j-1)\mathfrak{z}^+(i-1) + \mathfrak{b}_{\mathfrak{S}} + \mathcal{P}_{\mathfrak{S}}\mathfrak{a}'_{\mathfrak{S}}\right] \\
\epsilon_{\mathfrak{S}} &= \mathcal{T}(i) - H_{R\mathfrak{S}}\underline{\mathfrak{z}}_{\mathfrak{S}} \\
\nu_{\mathfrak{S}} &= \mathcal{D}^{-1}_{\mathfrak{S}}\epsilon_{\mathfrak{S}}
\end{aligned} \tag{31}$$

The recursion step from the i^{th} body to body $(j+1)$ is now as follows:

$$\begin{aligned}
\underline{\mathfrak{z}}^+_{\mathfrak{S}} &= \underline{\mathfrak{z}}_{\mathfrak{S}} + \underline{\mathcal{G}}_{\mathfrak{S}}\epsilon_{\mathfrak{S}} \\
\mathfrak{z}(i+1) &= E_{\mathfrak{S}}\underline{\mathfrak{z}}^+_{\mathfrak{S}} + \mathfrak{b}(i+1) + \mathcal{P}(i+1)\mathfrak{a}(i+1)
\end{aligned} \tag{32}$$

The base-to-tip accelerations sweeps steps also are altered as follows. First the steps from body $(i+1)$ to the i^{th} aggregate body:

$$\ddot{\theta}_{R\mathfrak{S}} = \nu_{\mathfrak{S}} - \underline{\mathcal{G}}_{\mathfrak{S}}{}^*E_{\mathfrak{S}}{}^*\alpha(i+1) \tag{33}$$

The step from the i^{th} aggregate body to body $(j-1)$ is as follows:

$$\begin{aligned}
\alpha^+(i-1) &= \mathcal{B}_{\mathfrak{S}}{}^*\alpha_{\mathfrak{S}} \\
\ddot{\theta}(i-1) &= \nu(i-1) - \mathcal{G}^*(i-1)\alpha^+(i-1) \\
\alpha(i-1) &= \alpha^+(i-1) + H^*(i-1)\ddot{\theta}(i-1) + \mathfrak{a}(i-1)
\end{aligned} \tag{34}$$

5.2 General expression for $X_{\mathfrak{S}}$

Now we look at the problem of obtaining expressions for $X_{\mathfrak{S}}$. When the constraint is directly among the joint angles, as for the geared link/motor case, $X_{\mathfrak{S}}$ is straightforward to write. When the constraint is a closure constraint as for the four-bar linkage case, the constraint can typically be expressed as:

$$Y\dot{\theta}_{\mathfrak{S}} = [Y_1,\ Y_2]\begin{bmatrix}\dot{\theta}^1_{\mathfrak{S}} \\ \dot{\theta}_{R\mathfrak{S}}\end{bmatrix} = \mathbf{0} \tag{35}$$

In the above, the above partition is such that Y_1 is square and full rank and so

$$\dot{\theta}^1_{\mathfrak{S}} = -Y_1^{-1}Y_2\dot{\theta}_{R\mathfrak{S}} \implies X_{\mathfrak{S}} = \begin{bmatrix}-Y_1^{-1}Y_2 \\ \mathbf{I}\end{bmatrix} \tag{36}$$

In the above, Y can be a constraint directly on the generalized velocities, or an indirect constraint on the link spatial velocities.

We next look into computing $\dot{X}_{\mathfrak{S}}$. required by Eq. (18). First we note that

$$\begin{aligned}
Y_1Y_1^{-1} = \mathbf{I} &\implies \frac{\mathbf{d}Y_1}{\mathbf{d}t}Y_1^{-1} + Y_1\frac{\mathbf{d}Y_1^{-1}}{\mathbf{d}t} = \mathbf{0} \\
&\implies \frac{\mathbf{d}Y_1^{-1}}{\mathbf{d}t} = -Y_1^{-1}\frac{\mathbf{d}Y_1}{\mathbf{d}t}Y_1^{-1}
\end{aligned} \tag{37}$$

Hence, with $Z \overset{\triangle}{=} Y_1^{-1}Y_2$,

$$\begin{aligned}
\frac{\mathbf{d}Z}{\mathbf{d}t} &= \frac{\mathbf{d}Y_1^{-1}}{\mathbf{d}t}Y_2 + Y_1^{-1}\frac{\mathbf{d}Y_2}{\mathbf{d}t} \overset{37}{=} -Y_1^{-1}\frac{\mathbf{d}Y_1}{\mathbf{d}t}Y_1^{-1}Y_2 + Y_1^{-1}\frac{\mathbf{d}Y_2}{\mathbf{d}t} \\
&= Y_1^{-1}\left[\frac{\mathbf{d}Y_2}{\mathbf{d}t} - \frac{\mathbf{d}Y_1}{\mathbf{d}t}Z\right]
\end{aligned}$$

Thus

$$\dot{X}_{\mathfrak{S}} = \begin{bmatrix} -\dot{Z} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} Y_1^{-1}\left[\dot{Y}_1 Z - \dot{Y}_2\right] \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} -Y_1^{-1}\dot{Y}X_{\mathfrak{S}} \\ \mathbf{0} \end{bmatrix} \tag{38}$$

Therefore

$$\dot{X}_{\mathfrak{S}}\dot{\theta}_{R\mathfrak{S}} = \begin{bmatrix} -Y_1^{-1}\dot{Y}X_{\mathfrak{S}}\dot{\theta}_{R\mathfrak{S}} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} -Y_1^{-1}\dot{Y}\dot{\theta}_{\mathfrak{S}} \\ \mathbf{0} \end{bmatrix} \tag{39}$$

5.3 Computing $\dot{Y}$

So far we have made no assumptions about the nature of Y. We look now at computing $\dot{Y}$ for the two predominant cases. The first more straightforward case is where there is an direct (often linear) algebraic relationship between the constrained generalized velocities. For this case $\dot{Y}$ is simply the product of gradient of this relationship with respect to to the sub-graph's generalized coordinates $\theta_{\mathfrak{S}}$ and the $\dot{\theta}_{\mathfrak{S}}$ sub-graph's generalized velocities. Geared joints are an example of this type of constraint where the gear ratio defines the constraint relationship.

The second situation is when the constraint consists of a algebraic constraint on the relative velocities of a pair of physical points in the sub-graph. Denoting these points as o and p, such a constraint can be expressed as

$$\begin{aligned} \mathbf{0} &= A(\mathcal{V}_o - \mathcal{V}_p) = A(\mathcal{J}_o - \mathcal{J}_p)\dot{\theta}_{\mathfrak{S}} \\ &= A\left[\mathcal{J}_{o1} - \mathcal{J}_{p1}, \quad \mathcal{J}_{o2} - \mathcal{J}_{p2}\right]\begin{bmatrix} \dot{\theta}^1_{\mathfrak{S}} \\ \dot{\theta}_{R\mathfrak{S}} \end{bmatrix} \end{aligned} \tag{40}$$

where $\mathcal{J}_o$, $\mathcal{J}_p$ denote Jacobians relating the sub-graph's generalized velocities to the spatial velocities at the points o and p, and $\mathcal{J}_{o1}$ etc. represents sub-blocks within these Jacobians. From the above relationship we can identify Y_1 and Y_2 as

$$Y_1 = A(\mathcal{J}_{o1} - \mathcal{J}_{p1}) \quad \text{and} \quad Y_2 = A(\mathcal{J}_{o2} - \mathcal{J}_{p2}) \tag{41}$$

Now lets examine how to derive an expressions for $\dot{Y}_1$ and $\dot{Y}_2$. From the chain rule, these derivatives require $\dot{A}$ and the time derivatives of the $\mathcal{J}_{o1}$ etc. blocks. The expression for $\dot{A}$ is usually straightforward to obtain from the nature of the constraint. So lets focus on the time derivative of the Jacobian blocks. Case in point, the i^{th} column of $\mathcal{J}_{o1}$ is of the form $\phi^*(\mathbb{O}_i, o)H^*(i)$, where $\mathbb{O}_i$ denotes the frame at the i^{th} hinge in $\dot{\theta}_{\mathfrak{S}}$ with $H^*(i)$ the corresponding joint map matrix at the hinge. The derivative involves the relative linear and angular velocity of o with respect to to $\mathbb{O}_i$ and is an easy expression to derive.

6 EXAMPLES OF CONSTRAINT EMBEDDING

Geared motors

For geared motors, we have the motor mounted on an inboard link driving the outer link that creates a closure-loop between the 3 bodies. For this sub-group we have:

$$\begin{aligned} &\dot{\theta}_{\mathfrak{S}} = [\theta_{mtr}(k),\, \theta_{lnk}(k)]^*, \quad \dot{\theta}_{R\mathfrak{S}} = \theta_{lnk}(k) \\ &X_{\mathfrak{S}} = [\mu_G(k),\, 1]^*, \quad \mathcal{B}_{\mathfrak{S}} = \mathfrak{B}_G(j), \quad H^*{}_{R\mathfrak{S}} = H_G{}^*(i) \\ &\mathcal{E}_{\phi_{\mathfrak{S}}} = \mathbf{0}, \quad \phi_{\mathfrak{S}} = \mathbf{I}, \quad E_{\mathfrak{S}} = \mathcal{A}_G(i+1, i) \end{aligned} \tag{42}$$

For this case, since $\phi_{\mathfrak{S}} = \mathbf{I}$, we can stop at Eq. (19) since the following equations become degenerately trivial. Using the definitions in Eq. (42) it can be verified that the expressions we obtain agree with those from earlier in this chapter.

Planar 4-bar linkage system (terminal cut)

Assume that the link a and c are directly connected to link $(i+1)$. Link b is the child of link a and its other end is connected to the end of link c through a hinge. Link $j-1$ is connected via a hinge to link b. The sub-graph consists of links a, b and c. We make a cut at the hinge joining links b and c to convert the sub-graph into a tree-topology system.

$$\begin{aligned} &E_{\mathfrak{S}} = [\mathbf{0},\, \mathbf{0},\, \phi(i+1, a)], \quad \mathcal{E}_{\phi_{\mathfrak{S}}} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \phi(b,c) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \phi(a,b) & \mathbf{0} \end{pmatrix} \\ &\phi_{\mathfrak{S}} = \begin{pmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \phi(b,c) & \mathbf{I} & \mathbf{0} \\ \phi(a,c) & \phi(a,b) & \mathbf{I} \end{pmatrix} \\ &H^*{}_{R\mathfrak{S}} = [X(c),\, X(b),\, H(a)]^*, \quad \mathcal{B}_{\mathfrak{S}}{}^* = [\mathbf{0},\, \mathbf{I},\, \mathbf{0}] \end{aligned} \tag{43}$$

In the above, $X(c)$ and $X(b)$ are the effective joint map matrices for the b and c matrices that satisfy the closed-loop wishbone constraint.

Planar 4-bar linkage system (internal cut)

Assume that the link a and c are directly connected to link $(i+1)$. Link b is the child of link a and its other end is connected to the end of link c through a hinge. Link $j-1$ is connected via a hinge to link b. The sub-graph consists of links a, b and c. We make a cut at the hinge joining links b and c to convert the sub-graph into a tree-topology system.

$$
\mathrm{E}_{\mathfrak{G}} = [\phi(i+1,c),\ \mathbf{0},\ \phi(i+1,a)], \quad \mathcal{E}_{\phi_{\mathfrak{G}}} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \phi(a,b) & \mathbf{0} \end{pmatrix}
$$

$$
\phi_{\mathfrak{G}} = \begin{pmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \phi(a,b) & \mathbf{I} \end{pmatrix}, \qquad \mathrm{H}^*{}_{\mathrm{R}\mathfrak{G}} = [\mathrm{X}(c),\ \mathrm{X}(b),\ \mathrm{H}(a)]^*
$$

$$
\mathcal{B}_{\mathfrak{G}}{}^* = [\mathbf{0},\ \mathbf{I},,\ \mathbf{0}] \tag{44}
$$

In the above, $\mathrm{X}(c)$ and $\mathrm{X}(b)$ are the effective joint map matrices for the b and c matrices that satisfy the closed-loop wishbone constraint.

7 CONCLUDING REMARKS

This paper has described a constraint embedding approach for the handling of local closure constraints in multibody system dynamics. The approach uses spatial operator techniques to eliminate local-loop constraints from the system and effectively convert the system into tree-topology systems. Once converted, the host of techniques available - including $\mathcal{O}(\mathcal{N})$ forward dynamics algorithms - are shown to be applicable to such systems. Future work will explore the implications of the sub-group definition when the system has a density of closely-coupled constraints. Another issue to be explored in more detail is the handling of singular configurations and consequent changes to the ranks of the constraint matrices.

Acknowledgments

The research described in this paper was performed at the Jet Propulsion Laboratory (JPL), California Institute of Technology, under contract with the National Aeronautics and Space Administration. The author would like to acknowledge the many insightful discussions with Dr. Rudranarayan Mukherjee on this and related multibody dynamics topics.

REFERENCES

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

[2] Rodriguez, G., 1987. "Kalman Filtering, Smoothing and Recursive Robot Arm Forward and Inverse Dynamics". IEEE Journal of Robotics and Automation, **3**(6), Dec., pp. 624–639.

[3] Featherstone, R., 1983. "The Calculation of Robot Dynamics using Articulated-Body Inertias". The International Journal of Robotics Research, **2**(1), Spring, pp. 13–30.

[4] Rodriguez, G., Jain, A., and Kreutz-Delgado, K., 1992. "Spatial Operator Algebra for Multibody System Dynamics". Journal of the Astronautical Sciences, **40**(1), Jan.–March, pp. 27–50.

[5] Critchley, R. J., and Anderson, K., 2003. "A generalized recursive coordinate reduction method for multibody dynamic systems". Journal of Multiscale Computational Engineering, **12**(2), pp. 181–200.

[6] Jain, A., and Rodriguez, G., 1992. "Recursive Flexible Multibody System Dynamics Using Spatial Operators". Journal of Guidance, Control and Dynamics, **15**(6), Nov., pp. 1453–1466.

[7] Luh, J., Walker, M., and Paul, R., 1980. "On-line Computational Scheme for Mechanical Manipulators". ASME Journal of Dynamic Systems, Measurement, and Control, **102**(2), June, pp. 69–76.

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

[9] Rodriguez, G., and Kreutz-Delgado, K., 1992. "Spatial Operator Factorization and Inversion of the Manipulator Mass Matrix". IEEE Transactions on Robotics and Automation, **8**(1), Feb., pp. 65–76.

[10] Rodriguez, G., 1990. "Random Field Estimation Approach to Robot Dynamics". IEEE Transactions on Systems, Man and Cybernetics, **20**(5), Sept., pp. 1081–1093.

[11] Jain, A., and Rodriguez, G., 2000. "Computational Robot Dynamics Using Spatial Operators". In IEEE International Conference on Robotics and Automation.

[12] Jain, A., and Rodriguez, G., 1993. "Linearization of Manipulator Dynamics Using Spatial Operators". IEEE Transactions on Systems, Man and Cybernetics, **23**(1), Jan., pp. 239–248.

[13] Jain, A., and Rodriguez, G., 1995. "Diagonalized Lagrangian Robot Dynamics". IEEE Transactions on Robotics and Automation, **11**(4), Aug., pp. 571–584.