Constraint Embedding for Multibody System Dynamics

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

Abstract

This paper describes 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. This approach allows the direct derivation of recursive $\mathsf{O}(\mathcal{N})$ *techniques for solving the system dynamics and avoiding the expensive steps that would otherwise be required for handling the closed-chain dynamics. The approach is very effective for systems where the constraints are confined to small-subgraphs within the system topology. The paper provides background on the spatial operator* $\mathsf{O}(\mathcal{N})$ *algorithms, the extensions for handling embedded constraints, and concludes with some examples of such constraints.*

There has been a considerable amount of interest in recent years in the development of efficient computational algorithms for solving the dynamics of multibody systems. The $\mathsf{O}(\mathcal{N})$ class of forward dynamics algorithms stand out as important breakthroughs on this front [1, 2, 3]. The $\mathsf{O}(\mathcal{N})$ algorithms apply to tree-topology multibody systems. Extensions to the broader class of closed-chain systems build upon the tree-topology algorithms. A closed-chain approach consists of treating the closed-chain topology as a tree-topology system subject additional closure constraint. The resulting forward dynamics solution consists of [**?**]: (a) ignoring the closure constraints and using the $\mathsf{O}(\mathcal{N})$ algorithm to solve for the "free" unconstrained accelerations for the system; (b) using the tree-topology solution to compute a correction force to enforce the closure-constraints; followed by (c) correcting the unconstrained accelerations with correction acclerations resulting from the correction forces. The correction step (b) required the computation of the reflected inertias at the closure nodes, referred to as the operational space inertia in order to obtain the correction forces. This is an expensive process requiring recursive computations across the system topology.

The subject of this paper on the class of multibody systems where the closure-constraints are local, i.e., where they are confined to small groupings of bodies within the system. Important examples of such local closure-constraints are constraints associated with four-bar linkages, geared motors, differential suspensions etc. This paper shows how one can elimnate these closure-constraints by embedding them directly into the system dynamics and effectively replacing the body groupings with virtual aggregate bodies. Once eliminated, we are able to extend the well-known $\mathsf{O}(\mathcal{N})$ algorithms to solve the dynamics of the system.

We begin with a brief overview of the spatial operator algebra approach to setting up the equations of motion for the system, followed by the steps leading to the derivation of $\mathsf{O}(\mathcal{N})$ 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 $\mathsf{O}(\mathcal{N})$

algorithm needed to handle the embedded constraints. We conclude with specific examples of embedding local closure-constraints.

1 Overview of Spatial Operators for Serial Chain Systems

The aim of this subsection is to summarize briefly 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 a much more general class of complex joint-connected mechanical systems, including tree configurations with flexible links and joints [4].

Consider a serial manipulator with $\mathcal{N}$ rigid links in Figure **??**. 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 kth 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 kth link. The kth 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 kth joint coordinate given by $\theta(k)$. Extension to joints with more rotational and translational degrees-of-freedom is easy [5].

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

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

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 kth 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 kth 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 kth joint is $H^*(k)\dot{\theta}(k)$. The spatial force of interaction $f(k)$ across the kth 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 kth link in the coordinate frame $\mathbb{O}_k$ is

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

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

The recursive Newton–Euler equations are [6, 2]

$$\begin{cases} & V(\mathcal{N}+1)=0; \qquad \alpha(\mathcal{N}+1)=0 \\ \textbf{for } k & = \mathcal{N}\cdots 1 \\ \quad V(k) & = \phi^*(k+1,k)V(k+1)+H^*(k)\dot{\theta}(k) \\ \quad \alpha(k) & = \phi^*(k+1,k)\alpha(k+1)+H^*(k)\ddot{\theta}(k)+\mathfrak{a}(k) \\ \textbf{end loop} & \end{cases}$$

$$\begin{cases} & f(0)=0 \\ \textbf{for } k & = 1\cdots\mathcal{N} \\ \quad f(k) & = \phi(k,k-1)f(k-1)+\mathbf{M}(k)\alpha(k)+\mathfrak{b}(k) \\ \quad \mathcal{T}(k) & = H(k)f(k) \\ \textbf{end loop} & \end{cases}$$

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 kth 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 [7] 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, 8]:

$$V = \phi^* H^* \dot{\theta}; \qquad \alpha = \phi^*[H^*\ddot{\theta} + \mathfrak{a}] \tag{1}$$

$$f = \phi[\mathbf{M}\alpha + \mathfrak{b}]; \qquad \mathcal{T} = Hf = \mathcal{M}\ddot{\theta} + \mathcal{C} \tag{2}$$

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 $\phi \in \mathcal{R}^{6\mathcal{N}\times 6\mathcal{N}}$

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

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

$$\mathcal{E}_\phi = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ \phi(2,1) & 0 & \dots & 0 & 0 \\ 0 & \phi(3,2) & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \phi(\mathcal{N},\mathcal{N}-1) & 0 \end{pmatrix} \tag{4}$$

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

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

These identifies have been used extensively [2, 8, 9, 7, 3, 10, 11, 12], 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 [1].

1.1 Mass Matrix Innovations Factorization

The mass matrix has [**?**] the alternative factorization

$$\mathcal{M} = (\mathsf{I} + \mathsf{H}\phi\mathsf{K})\mathsf{D}(\mathsf{I} + \mathsf{H}\phi\mathsf{K})^* \tag{5}$$

where the outer factors are mutual transposes of each other. The factor $(\mathsf{I} + \mathsf{H}\phi\mathsf{K})$ is a square, invertible matrix whose inverse is

$$(\mathsf{I} + \mathsf{H}\phi\mathsf{K})^{-1} = \mathsf{I} - \mathsf{H}\psi\mathsf{K} \tag{6}$$

involving the articulated Kalman filtering recursion ψ and the shifted Kalman gain operator K. This implies that the inverse of the mass matrix is

$$\mathcal{M}^{-1} = (\mathsf{I} - \mathsf{H}\psi\mathsf{K})^*\mathsf{D}^{-1}(\mathsf{I} - \mathsf{H}\psi\mathsf{K}) \tag{7}$$

The identity implies, and is implied by, a tip-to-base Kalman filtering operation followed by a base-to-tip Bryson smoothing operation. The spatially recursive algorithm that results has been shown to be equivalent to the articulated inertia forward dynamics algorithm advanced by Featherstone [1]. The identity also results in an explicitly symbolic expression for the inverse of the mass matrix.

2 $O(\mathcal{N})$ Forward Dynamics

The expression in Eq. (??) can be broken into the following sequence of intermeidiate quantities:

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

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 described in Algorithm ?? and its structure is illustrated in Figure ??.

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

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

This algorithm includes the recursive steps for the computation of the $\mathcal{P}(.)$'s and $\mathfrak{z}(.)$'s in Chapter ??. One modification here is to the $\mathfrak{z}(.)$'s part in that now we no longer have the restriction that the system is at rest. 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 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.

3 Constraint Embedding

Let us assume that within the multibody system, we we have a a sub-group of bodies with some constraints among them. In effect, the true degrees of freedom associated with this sub-group is less than the number associated with their collective hinge degrees of freedom . Examples of such local loops include those from geared motors, 4-bar linkages/wishbone suspensions, differentials, skid-steered wheels (eg. ATRVJr) etc. The question, we address here is the process of embedding these constraints directly into the dynamics model so that we can continue to use the simpler tree topology model instead of the more general closed-chain dynamics formulations.

The approach here is to think of replacing the subgroup with a single equivalent link so that the closure constraints are eliminated from the various recursions. 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 links to the system dynamics. Note, that the new model is an exact replacement for the original model and no approximations are involved.

The proposed approach requires that with appropriate renumbering, we can indeed number the links in the sub-group contiguously. Let us assume that the indexing goes from i to j with $i > j$.

Define $\mathcal{V}_{\mathfrak{S}} = \text{col}\left\{\mathcal{V}(i), \cdots \mathcal{V}(j)\right\}$, $\dot{\theta}_{\mathfrak{S}} = \text{col}\left\{\dot{\theta}(i), \cdots \dot{\theta}(j)\right\}$. Then we have,

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

With $\phi_{\mathfrak{S}} \triangleq (\mathbf{I} - \mathcal{E}_{\phi_{\mathfrak{S}}})^{-1}$, we have

$$\mathcal{V}_{\mathfrak{S}} = \phi^*{}_{\mathfrak{S}}E_{\mathfrak{S}}{}^*\mathcal{V}(i+1) + \phi^*{}_{\mathfrak{S}}H^*{}_{\mathfrak{S}}\dot{\theta}_{\mathfrak{S}} \tag{11}$$

Note that $\phi_{\mathfrak{S}}$ is a subblock 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 aggregate link. Our new aggregate link for the sub-graph then has spatial velocity defined by $\mathcal{V}_{\mathfrak{S}}$. Due to the internal constraints, clearly not all elements of $\dot{\theta}_{\mathfrak{S}}$ are independent. Hence, there exists a (configuration dependent) mapping $X_{\mathfrak{S}}$ such that

$$\dot{\theta}_{\mathfrak{S}} = X_{\mathfrak{S}}\dot{\theta}_{R\mathfrak{S}} \implies H^*{}_{\mathfrak{S}}\dot{\theta}_{\mathfrak{S}} = H^*{}_{R\mathfrak{S}}\dot{\theta}_{R\mathfrak{S}} \quad \text{where} \quad H^*{}_{R\mathfrak{S}} \triangleq H^*{}_{\mathfrak{S}}X_{\mathfrak{S}} \tag{12}$$

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. Hence, we have

$$\mathcal{V}_{\mathfrak{S}} = \phi^*{}_{\mathfrak{S}}E_{\mathfrak{S}}{}^*\mathcal{V}(i+1) + \phi^*{}_{\mathfrak{S}}H^*{}_{R\mathfrak{S}}\dot{\theta}_{R\mathfrak{S}} \tag{13}$$

With this setup, all of the inverse and forward dynamics results continue to hold for the new operators defined by embedding the constraints.

In the forward dynamics algorithm we thus have.

$$
\begin{aligned}
\mathcal{P}^{+}(j-1) &= \overline{\tau}(j-1)\mathcal{P}(j-1) \\
\mathcal{P}_{\mathfrak{S}} &= \mathcal{B}_{\mathfrak{S}}\phi(j,j-1)\mathcal{P}^{+}(j-1)\phi^{*}(j,j-1)\mathcal{B}_{\mathfrak{S}}{}^{*} + M_{\mathfrak{S}} \\
\mathcal{D}_{\mathfrak{S}} &= H_{R\mathfrak{S}}\phi_{\mathfrak{S}}\mathcal{P}_{\mathfrak{S}}\phi^{*}{}_{\mathfrak{S}}H^{*}{}_{R\mathfrak{S}} \\
\mathcal{G}_{\mathfrak{S}} &= \mathcal{P}_{\mathfrak{S}}\phi^{*}{}_{\mathfrak{S}}H^{*}{}_{R\mathfrak{S}}\mathcal{D}_{\mathfrak{S}}^{-1} \\
\tau_{\mathfrak{S}} &= \mathcal{G}_{\mathfrak{S}}H_{R\mathfrak{S}}\phi_{\mathfrak{S}} \\
\mathcal{P}^{+}_{\mathfrak{S}} &= \mathcal{P}_{\mathfrak{S}} - \tau_{\mathfrak{S}}\mathcal{P}_{\mathfrak{S}} \\
\mathcal{P}(i+1) &= E_{\mathfrak{S}}\mathcal{P}^{+}_{\mathfrak{S}}E_{\mathfrak{S}}{}^{*} + M(i+1)
\end{aligned}
\tag{14}
$$

Note that $\mathcal{D}_{\mathfrak{S}}$ has the structure of a mass matrix. This mass matrix is for the sub-graph multibody system. In this multibody system, link $(i+1)$ is regarded as the inertial frame, and so the bodies in the sub-graph directly attached to this link are assumed to be independent base-bodies attached to the inertial frame. Due to the cuts, this multi-body system has a tree-topology structure. The mass of the link j includes the articulated body contribution from the sub-tree rooted at link $j-1$. Since $\mathcal{D}_{\mathfrak{S}}$ is a mass matrix, $\mathcal{D}_{\mathfrak{S}}^{-1}$ can be factored using the Innovations factorization, i.e.

$$
\mathcal{D}_{\mathfrak{S}}^{-1} = [\mathbf{I} - H_{R\mathfrak{S}}\psi_{\mathfrak{S}}\mathcal{K}_{\mathfrak{S}}]^{*}\mathcal{D}_{I\mathfrak{S}}^{-1}[\mathbf{I} - H_{R\mathfrak{S}}\psi_{\mathfrak{S}}\mathcal{K}_{\mathfrak{S}}] \tag{15}
$$

The articulated body inertia quantities are ones for just the sub-graph. The $\mathcal{D}_{I\mathfrak{S}}^{-1}$ is the $\mathcal{D}^{-1}$ quantity associated with just the sub-graph multibody system as described above. Hence,

$$
\begin{aligned}
\mathcal{G}_{\mathfrak{S}} &= \mathcal{P}_{\mathfrak{S}}\psi^{*}{}_{\mathfrak{S}}H^{*}{}_{R\mathfrak{S}}\mathcal{D}_{I\mathfrak{S}}^{-1}[\mathbf{I} - H_{R\mathfrak{S}}\psi_{\mathfrak{S}}\mathcal{K}_{\mathfrak{S}}] \\
\tau_{\mathfrak{S}} &= \mathcal{P}_{\mathfrak{S}}\psi^{*}{}_{\mathfrak{S}}H^{*}{}_{R\mathfrak{S}}\mathcal{D}_{I\mathfrak{S}}^{-1}H_{R\mathfrak{S}}\psi_{\mathfrak{S}} = \mathcal{P}_{\mathfrak{S}}\Omega_{\mathfrak{S}} \\
\mathcal{P}^{+}_{\mathfrak{S}} &= \mathcal{P}_{\mathfrak{S}} - \mathcal{P}_{\mathfrak{S}}\Omega_{\mathfrak{S}}\mathcal{P}_{\mathfrak{S}}
\end{aligned}
\tag{16}
$$

Note that $\Omega_{\mathfrak{S}} \triangleq \psi^{*}{}_{\mathfrak{S}}H^{*}{}_{R\mathfrak{S}}\mathcal{D}_{I\mathfrak{S}}^{-1}H_{R\mathfrak{S}}\psi_{\mathfrak{S}}$ is the Operational Space Inertia matrix for the sub-graph, and that $\mathcal{P}_{\mathfrak{S}}\Omega_{\mathfrak{S}}$ is a projection operator.

Geared motors

$$
\begin{aligned}
&\dot{\theta}_{\mathfrak{S}} = [\theta_{mtr}(k),\ \theta_{lnk}(k)]^{*}, \quad \dot{\theta}_{R\mathfrak{S}} = \theta_{lnk}(k), \quad X_{\mathfrak{S}} = [\mu_{G}(k),\ \mathbf{I}]^{*}, \\
&\mathcal{E}_{\phi\mathfrak{S}} = \mathbf{0}, \quad \phi_{\mathfrak{S}} = \mathbf{I}, \quad E_{\mathfrak{S}} = \mathcal{A}_{G}(i+1,i), \quad H^{*}{}_{R\mathfrak{S}} = H_{G}{}^{*}(i), \quad \mathcal{B}_{\mathfrak{S}} = \mathfrak{B}_{G}(j)
\end{aligned}
\tag{17}
$$

For this case, since $\phi_{\mathfrak{S}} = \mathbf{I}$, we can stop at Eq. (14) since the following equations become degenerately trivial. Using the definitions in Eq. (17) it can be verified that the expressions we obtain agree with those from earlier in this chapter.

Planar 4-bar linkage wishbone system

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 subgraph 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.

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

$$\phi_{\mathfrak{S}} = \begin{pmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \phi(a,b) & \mathbf{I} \end{pmatrix}, \qquad H^*{}_{R\mathfrak{S}} = [X(c),\ X(b),\ H(a)]^*, \quad \mathcal{B}_{\mathfrak{S}}{}^* = [\mathbf{0},\ \mathbf{I},\ \mathbf{0}] \tag{18}$$

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.

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 wishbone case, it can typically be expresed as:

$$Y\dot{\theta}_{\mathfrak{S}} = [Y_1,\ Y_2] \begin{bmatrix} \dot{\theta}_{\mathfrak{S}1} \\ \dot{\theta}_{\mathfrak{S}2} \end{bmatrix} = \mathbf{0} \tag{19}$$

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

$$\dot{\theta}_{\mathfrak{S}1} = -Y_1^{-1}Y_2\dot{\theta}_{\mathfrak{S}2} \implies X_{\mathfrak{S}} = \begin{bmatrix} -Y_1^{-1}Y_2 \\ \mathbf{I} \end{bmatrix} \tag{20}$$

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

tSubgraph

4 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 $O(\mathcal{N})$ forward dynamics algorithms - are shown to be applicable to such systems.

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

[2] 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.

[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] 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.

[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] 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.

[7] 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.

[8] 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.

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

[10] A. Jain and G. Rodriguez, "Computational Robot Dynamics Using Spatial Operators," in IEEE International Conference on Robotics and Automation, (San Francisco), Apr. 2000.

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

[12] A. Jain and G. Rodriguez, "Diagonalized Lagrangian Robot Dynamics," IEEE Transactions on Robotics and Automation, vol. 11, pp. 571–584, Aug. 1995.