A LOGARITHMIC COMPLEXITY DIVIDE-AND-CONQUER ALGORITHM FOR MULTI-FLEXIBLE ARTICULATED BODY DYNAMICS

Rudranarayan M. Mukherjee[⋆], **Kurt S. Anderson**[⋆]

[⋆]Computational Dynamics Laboratory
Department of Mechanical Aerospace and Nuclear Engineering
Rensselaer Polytechnic Institute
110 8th Street. Troy NY 12180 USA
e-mail: mukher@rpi.edu, anderk5@rpi.edu

Keywords: Flexible Body Dynamics, Logarithmic Computational Complexity, Divide and Conquer.

Abstract. *This paper presents an efficient algorithm for the parallel implementation of dynamics simulation and analysis of multi-flexible-body systems. This algorithm formulates and solves the nonlinear equations of motion for mechanical systems with interconnected flexible bodies subject to the limitations of modal superposition, and body substructuring, with arbitrarily large rotations and translations. The large rotations or translations are modelled as rigid body degrees of freedom associated with the interconnecting kinematic joint degrees of freedom. The elastic deformation of the component bodies is modelled through the use of modal coordinates and associated admissible shape functions. Apart from the approximation associated with the elastic deformations, this algorithm is exact, non-iterative and applicable to generalized multi-flexible chain and tree topologies. In its basic form, the algorithm is both time and processor optimal in its treatment of the n_b joint variables, providing $O(log(n_b))$ turn around time per temporal integration step, achieved with $O(n_b)$ processors. The actual cost associated with the parallel treatment of the n_f flexible degrees of freedom depends on the specific parallel method chosen for dealing with the inversion of the individual coefficient matrices which are associated locally with each flexible body.*

$\vec{r}$	Position vector
V	Spatial velocity, a 6×1 column matrix
A	Spatial acceleration, a 6×1 column matrix
$\vec{\omega}$	Angular velocity
$\vec{v}$	Translational velocity of a body reference point
$\vec{\alpha}$	Angular acceleration
$\vec{a}$	Translational acceleration of a body reference point
$\vec{\varphi}$	Admissible shape function for translational components of deformation
$\vec{\psi}$	Admissible shape function for rotational components of deformation
Φ	Spatial matrix containing the shape functions $\begin{bmatrix}\vec{\psi}\\ \vec{\varphi}\end{bmatrix}$
${q_i}^k$	i-th Modal Coordinate of a flexible body k
$\dot{q}_i^k$	Time derivative of i-th modal coordinate of a flexible body k
$\ddot{q}_i^k$	Second time derivative of i-th modal coordinate of a flexible body k
ρ	Mass density
ζ	Inertia coupling terms for individual body or subassembly
Υ	Inertia coupling terms for assembly
n_b	Number of bodies in the system
ν^k	Number of assumed mode shapes for a flexible body k
$H^{k/k+1}$	Joint motion subspace map of joint connecting body k+1 to body k
$D^{k/k+1}$	Orthogonal complement of $H^{k/k+1}$
u	Generalized relative speed
$\dot{u}$	Time derivative of generalized relative speed
${}^N\vec{\omega_r}^{k^+}$	r-th partial angular velocity of body k w.r.t frame N
${}^N\vec{v_r}^{k^+}$	r-th partial velocity of body k w.r.t frame N
P_r^k	r-th spatial partial velocity of body k w.r.t frame N
$S^{k/k+1}$	Shift Matrix between joint k and k+1 $\begin{bmatrix}\underline{U} & \vec{r}\times\\ \underline{0} & \underline{U}\end{bmatrix}$
$\vec{b}\times$	3×3 skew symmetric matrix for cross product of any vector $\vec{b}$
C^T	Transpose of any arbitrary matrix C
${F_c}^i$	Spatial constraint force at joint *i*
${\vec{\tau_c}}^i$	Constraint torque at joint *i*
${\vec{f_c}}^i$	Constraint force at joint *i*
${\tilde{\tau_c}}^i$	Measure numbers of constraint torque at joint *i*
${\tilde{f_c}}^i$	Measure numbers of constraint force at joint *i*
$\vec{f^p}$	Body force at point *p*
$\hat{k_i}$	Unit vector in direction *i*
$\underline{\mathrm{U}}$	3×3 Identity matrix
$\underline{0}$	3×3 Zero matrix
$\vec{\varphi}\|_P$	$\vec{\varphi}$ evaluated at point P

Table 1: The Nomenclature

1 Introduction

Modelling and simulation of the dynamic behavior of complex systems are regularly pursued by engineers and scientists in a wide variety of fields. Applications may include coarse-grained molecular dynamics simulations for advanced material modelling (e.g. polymer chains) and biomolecular systems (e.g. RNA, DNA and proteins); elastic deformation of MEMS devices; modelling the dynamic behavior of multi-continuous bodies (e.g. drive belts, tracks and tracked vehicles); robotic systems and myriad mechanisms and electro-mechanical devices. These systems are typically modelled as articulated, i.e. a system comprising of rigid and (or) flexible bodies interconnected by kinematic joints to form a chain, tree or kinematically closed loop topology. Depending on the system considered and the resolution of the model, these simulations may include a large number of spatial degrees of freedom. For example, applications such as biomolecular systems or molecular modelling of materials may easily involve several thousand ($> 10^5$) spatial degrees of freedom and aim to capture phenomenon over large temporal scales ($> 10^6$ temporal integration steps). With the continued trend towards ever increasing problem size (greater model fidelity, and larger scale systems) and hence growing computational burden, use of efficient algorithms and exploring parallel computing resources have emerged as the primary means to reduce simulation turn-around times.

Due to this accelerating need for understanding, predicting, and controlling the dynamic behavior of many modern engineering systems, the development of algorithms for modelling the dynamic behavior of multibody systems has received considerable attention over the past three decades. These efforts have spawned algorithms and implementation procedures based on fundamentally different philosophies, each with their own strengths and weaknesses. Some of the earliest algorithms were $O(n^3)$ complexity [1]-[2], indicating that the computational cost, which manifests itself in computer CPU time, increases as a cubic function of the number of system generalized coordinates n. Subsequently several algorithms [3]-[9] were independently proposed by various authors for solving the articulated rigid body dynamics problem in $O(n)$ complexity. A brief review of some of these and a discussion of the underlying similarities between different algorithms has been discussed in [9]. These algorithms were initially limited to multi-rigid body applications, but were then extended and generalized to accommodate flexible body systems [10]-[15]. With these so-called $O(n)$ algorithms, the simulation turn around times scale approximately linearly with the increase in system size and hence are more efficient than the traditional $O(n^3)$ approach when applied to large (n$>>$1) systems.

The lowest computational order possible for articulated body systems when using serial processing is $O(n)$. Parallel processing offers some potential for improving on this. With parallel processing, it becomes theoretically possible to generate and solve the equations of motion of the system in as low as $O(log(n))$ complexity. Actual performance is generally not able to achieve the theoretical complexity due to restrictions of true parallel performance as described by Amdahl's law [16], and actual inter-processor communications costs.

In 1995 Fijany and Sharf [17] proposed the *Constraint Force Algorithm (CFA)* for serial kinematic chains. This was the first parallel algorithm which was time optimal, i.e. $O(log(n))$ in time per temporal integration step and processor optimal, i.e. $O(log(n))$ performance achieved with only $O(n)$ processors. In 1999 Featherstone [18]-[19] proposed a Divide and Conquer Algorithm for articulated rigid body systems (hereafter referred to as RDCA). This algorithm also achieved time optimal $O(log(n))$ complexity for parallel implementation with processor optimal $O(n)$ processors, and could be applied to more general topologies. This algorithm is highly efficient for simulating the forward dynamics of articulated rigid body systems when

implemented in parallel on number of processors equal to or greater than the number of bodies in the system.

Rigid body dynamics, though able to represent the gross behavior of many systems, is often inadequate in capturing the essential behavior of systems with flexible bodies. In this paper, we present a Divide and Conquer Algorithm for flexible bodies. This algorithm is a generalization of the RDCA to include flexible bodies in the articulated system and maintains logarithmic computational complexity. The goal of this algorithm is to produce and subsequently solve the equations of motion for multi-flexible body systems such that the equations associated with the individual bodies are effectively uncoupled from those of the other system bodies, and are also highly parallelizable.

The individual bodies that form the articulated system are modelled as flexible subject to the limitations of modal superposition and body substructuring, with arbitrarily large rotations and translations. A component mode type formulation is used whereby a reduced set of assumed modes is used to describe the deformation of a component body. These assumed mode shapes could be free-free, clamped-free, or any other appropriate set of vibration modes, constraint modes and / or any required static correction modes. The mode shapes can be obtained for each body from finite element analysis, analytic models or experimental analysis. The choice of the reduced set of shapes has been studied in depth in literature [20]-[21]-[22] and without loss of generality it is assumed that a set of admissible mode shapes is obtained for each component body as an input to this algorithm. The temporal coefficients of these mode shapes are treated as the flexible degrees of freedom and solved for in this algorithm. The coupling between the finite joint rotations and the elastic deformations of the bodies is preserved in this formulation. Consequently, the mass matrix obtained for each flexible body is a nonlinear function of the joint and flexible body generalized coordinates associated locally with it, while the bias force terms are nonlinear functions of the generalized coordinates and velocities [23].

2 Theoretical Development of the Algorithm

The theoretical development which follows is divided into two sections. The first deals with the basic divide and conquer algorithm which involves the interactions of assemblies with other assemblies through their connecting handles (joints). These assemblies may be point masses, rigid bodies, flexible bodies, or collections of any of these. As such this first portion of the development is independent of the type of components which comprise the assemblies. The subsequent section of the development is dedicated to the efficient handling of aspects which are specific to the treatment of flexible bodies within these assemblies.

2.1 Basic Divide and Conquer Algorithm

The basic algorithm works in a manner highly similar to the DCA scheme explained in detail in [18]-[19]. It is presented here so that this paper might be more self contained. The basic unit of the DCA scheme is the two-handle representation of a body. A handle is any selected reference on the body through which the interactions of the body with the environment can be modelled. The handles on a body can correspond to a joint location, a center of mass or any desired reference point. The two handles can even coincide. For the algorithm presented here, the joint locations are chosen as the handles on the body. Consider two representative bodies *Body k* and *Body k+1* of the articulated body system as shown in figure (1). The two handles on *Body k* correspond to the Joints J^{k+} and J^{k+1^-}. Similarly, the two handles on *Body k+1* correspond to the joints J^{k+1^+} and J^{k+2^-}.

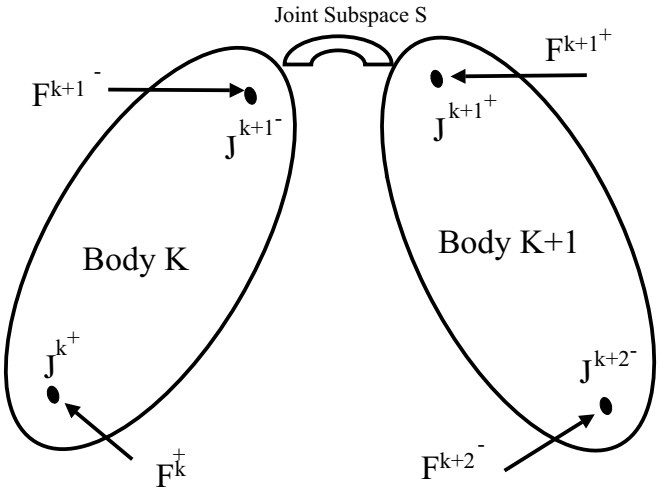

Figure 1: Two Handle Articulated Body

There are two main processes in the DCA approach, a hierarchic assembly process and a hierarchic disassembly process. In the hierarchic assembly process, the equations of motion of each body are written in terms of the accelerations occurring at each of its two handles (the procedure may be easily generalized to bodies with more than two handles) in the form expressed below for a representative body *Body k*.

$$A^{k^+} = \zeta_{11}{}^k F_c^{k+} + \zeta_{12}{}^k F_c^{k+1^-} + \zeta_{13}{}^k \quad (1)$$

$$A^{k+1^-} = \zeta_{21}{}^k F_c^{k+} + \zeta_{22}{}^k F_c^{k+1^-} + \zeta_{23}{}^k \quad (2)$$

The above equation set is henceforth referred to as the two handle equations of motion of representative body *Body k*. Here A^{k^+} and A^{k+1^-} are the spatial accelerations of *Body k* at joint locations J^{k+} and J^{k+1^-} respectively. The modal coordinates, as well as their first and second time derivatives manifest themselves in the kinematic relationship which relate accelerations of each of the body's handles. This is shown in detail in section (3.2) which presents the derivation of the equations of motion. The terms $\zeta_{ij}{}^k \, (i,j = 1,2)$ correspond to inertia coupling at the joint locations on *Body k*. F_c^{k+} and $F_c^{k+1^-}$ are the unknown spatial constraint loads acting on the body at the joints. The active forces at the joint are state dependent and are treated as known quantities. These are lumped together with the state dependent inertia forces and expressed as ζ_{ij} (i=1,2 j=3). Similarly the two handle equations of motion for *Body k+1* can be written in the form

$$A^{k+1^+} = \zeta_{11}{}^{k+1} F_c^{k+1^+} + \zeta_{12}{}^{k+1} F_c^{k+2^-} + \zeta_{13}{}^{k+1} \quad (3)$$

$$A^{k+2^-} = \zeta_{21}{}^{k+1} F_c^{k+1^+} + \zeta_{22}{}^{k+1} F_c^{k+2^-} + \zeta_{23}{}^{k+1} \quad (4)$$

In the hierarchic assembly process, the two handle equations of motion of two successive bodies are coupled together to form the two handle equations of motion of the resulting assembly

$$A^{k^+} = \Upsilon_{11} F_c^{k+} + \Upsilon_{12} F_c^{k+2^-} + \Upsilon_{13} \quad (5)$$

$$A^{k+2^-} = \Upsilon_{21} F_c^{k+} + \Upsilon_{22} F_c^{k+2^-} + \Upsilon_{23} \quad (6)$$

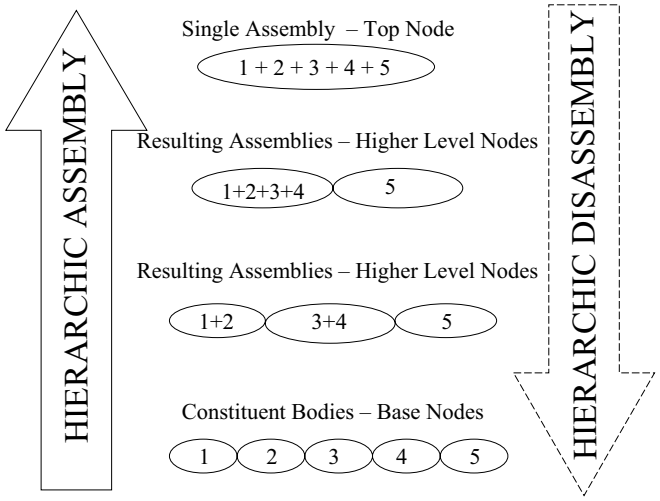

Figure 2: The Hierarchic Assembly and Disassembly Process using Binary Tree Structure

The two handles of the resulting assembly are the inward joint of the *Body k* viz. J^{k+} and the outward joint on the *Body k+1* viz. J^{k+2^-} and the constraint forces are those acting on the resulting assembly at those handles. The inertia coupling terms, Υ_{ij}, for the resulting assembly are calculated using a recursive set of formulae as discussed and presented in section (3.3) of this paper.

This process begins at the level of individual bodies of the system. Adjacent bodies of the system are hierarchically assembled as the construction of a binary tree as shown in figure (2). Individual bodies that make up the system form the base nodes of the binary tree. The equations of motion of a pair of bodies are coupled together using the recursive set of formulae to form the two handle equations of motion of the resulting assembly. The resulting assembly now corresponds to a node of the next level in the binary tree. Working up the binary tree in this hierarchic assembly processes, only a single assembly is left as the top node of the binary tree. The top node corresponds to the two-handle representation of the entire articulated system modelled as a single assembly. The two handles on this body correspond the boundary joints of the articulated system.

If the system is free floating, the constraint forces on the two handles are identically zero. The spatial accelerations can be easily obtained as

$$A^{k+} = \Upsilon_{13} \quad \text{and} \quad A^{k+2^-} = \Upsilon_{23} \tag{7}$$

If the system is anchored then the constraint force F_c^{k+} on the inward anchored joint is non-zero while at the free end, $F_c^{k+2^-}$ is zero. However the constraint force F_c^{k+} lies in the subspace orthogonal to the motion subspace of the joint. Hence the dot product of the matrix $H^{k/(k+1)}$ which maps the joint motion subspace and the constraint force F_c^{k+} is identically zero. The joint motion subspace $H^{k/(k+1)}$ and its orthogonal complement $D^{k/(k+1)}$ are discussed in further detail in section (3.1) associated with the kinematic preliminaries. Thus, using the joint motion subspace matrix, the two handle equations of motion of the top node can be manipulated as

shown below.

$$A^{k^+} = H^{k/(k+1)}\dot{u} + \dot{H}^{k/(k+1)}u = \Upsilon_{11}F_c^{k+} + \Upsilon_{13} \tag{8}$$

$$\Rightarrow H^{k/(k+1)^T}{\Upsilon_{11}}^{-1}(H^{k/(k+1)}\dot{u} + \dot{H}^{k/(k+1)}u - \Upsilon_{13}) = H^{k/(k+1)^T}F_c^{k+} = 0 \tag{9}$$

$$\Rightarrow \quad H^{k/(k+1)^T}{\Upsilon_{11}}^{-1}H^{k/(k+1)}\dot{u} = H^{k/(k+1)^T}{\Upsilon_{11}}^{-1}(\Upsilon_{13} - \dot{H}^{k/(k+1)}u) \tag{10}$$

$$\text{Let} \quad \mathrm{Q} = [\mathrm{H}^{\mathrm{k/(k+1)^T}}{\Upsilon_{11}}^{-1}\mathrm{H}^{\mathrm{k/(k+1)}}] \tag{11}$$

$$\Rightarrow \dot{u} = Q^{-1}H^{k/(k+1)^T}{\Upsilon_{11}}^{-1}(\Upsilon_{13} - \dot{H}^{k/(k+1)}u) \tag{12}$$

In the above equations Υ_{11} and Q are symmetric positive definite matrices (SPD) and there is no problem with the matrix inversion. The term $\dot{H}^{k/(k+1)}u$ is state dependent and is easily calculated. Having solved for $\dot{u}$, the spatial acceleration A^{k^+} can be obtained by equation (5). This value of the spatial acceleration is substituted into the two handle equations of motion at the top node to generate the constraint force and spatial acceleration at the other handle. Thus, whether the system is free floating or anchored, the equations of motion at the top node can be solved to generate the values of the spatial accelerations and constraint forces at the two handles. The case where the system contains kinematically closed loops is the topic of an associated forthcoming paper.

The hierarchic disassembly process begins with the solution of the two-handle equations of motion of the top node. From this solution, the accelerations of and forces on the two handles of the single assembly are known. The spatial acceleration and constraint forces generated by solving the two handle equations of an assembly are identically the values of the spatial accelerations and constraint forces on one handle on each of the two constituent assemblies. From these known quantities, the two handle equations of motion of the constituent assemblies can be easily solved for the constraint force and spatial acceleration at the connecting joint. For example, for a representative assembly made from *Body k* and *Body k+1*, the equations of motion are given by equation (5-6). On solving these equations the quantities A^{k^+}, A^{k+2^-}, $F_c^{k^+}$ and $\mathrm{F_c^{k+2^-}}$ are generated. These quantities are then substituted into the two-handle equations of the constituent sub-assemblies say for *Body k* and *Body k+1*. Thus knowing the values of A^{k^+}, $F_c^{k^+}$, equations (1-2) can be easily solved, while from A^{k+2^-} and $F_c^{k+2^-}$ equations (3-4) can be solved. This process is repeated in a hierarchic disassembly of the binary tree where the known boundary conditions are used to solve the two-handle equations of motion of the immediate subassemblies, until spatial acceleration and constraint forces on all bodies in the system are calculated. As mentioned above, the time derivatives of the modal coordinates are embedded in the kinematics which relate spatial accelerations of the joints on each individual body. Having solved for the spatial accelerations of the joints in the hierarchic disassembly, the time derivatives of the modal coordinates can now be solved for independently on each body. The expression for the time derivatives of the modal coordinates is derived in detail in the section (3.2) which deals with the derivation of the equations of motion.

Similar to the scheme in RDCA, this algorithm works in four sweeps, traversing the system topology like a binary tree. The first and the third sweep work upwards from the base nodes of the binary tree to the top node while the second and the fourth sweep work downwards. The input to this algorithm is comprised of the mass properties of the bodies, joint generalized coordinates and speeds, the modal coordinates and their first time derivatives as well as the admissible shape functions for each body. The first two sweeps generate the position and velocity of each handle on each node by using an assembly-disassembly process similar to that described in [18]. On completing the two sweeps, on each base node, the coordinate transformations, the

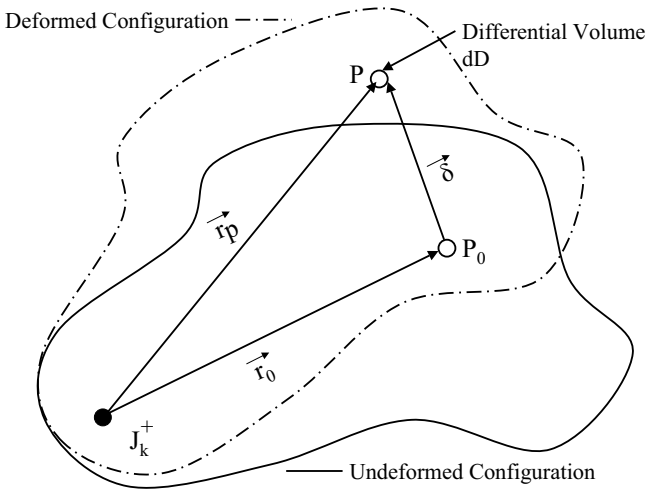

Figure 3: Deformed and Undeformed Configuration of Representative Body k

state dependent accelerations, the active joint forces and the state dependent elastic deformation terms are calculated. The active forces are state dependent and include actuator forces on the joints, damping forces, body forces like gravity as well as the elastic forces arising from the deformation of the bodies. The final two sweeps correspond to the hierarchic assembly and the hierarchic disassembly processes respectively.

In the analytical treatment presented here, direction cosine matrices and transformation between different basis are not shown explicitly. Appropriate basis transformations have to be taken into account for an implementation of this algorithm. Also, this algorithm uses a mixed set of coordinates viz. Cartesian coordinates and relative coordinates throughout the derivation. Mixed set of coordinates has been used in [24]-[25] for rigid body dynamics and in [26] for flexible body dynamics among others.

3 Extension to Flexible Bodies

The following subsections present the details associated with efficiently dealing with flexible bodies within the divide and conquer assemblies, and having these flexible aspects manifest themselves properly through the assembly handles to the rest of the system.

3.1 Kinematic Preliminaries

Consider flexible *Body k* as a typical body of the articulated body system. *Body k* has two handles corresponding to the two joints by which it is connected to the rest of the system. As with the basic development presented above, this procedure is easily extendable to more than two handles. As before, the inward joint is J^{k+} and the outward joint is J^{k+1^-}. The local "body fixed" reference frame associated with $Body\ k$, is rigidly attached to the material point of $Body\ k$, located at joint J^{k^+}, and thus rotates and translates with that material point. *Body k* undergoes relative rigid body motion at the joint J^{k+} with respect to its inward (parent) body and deforms elastically with respect to its body fixed reference frame.

Figure (3) shows the deformed and undeformed configurations of the *Body k*. In its undeformed configuration let P_0 be an arbitrary differential volume δD and let the vector $\vec{r_0}$ describe the position of P_0 with respect to the body fixed reference frame at J^{k+}. After undergoing elas-

tic deformation, let P denote the differential volume in the deformed configuration. $\vec{r_p}$ is the vector describing the position of point P with respect to J^{k+} expressed in its body basis. The displacement vector $\vec{\delta}$ is expressed in terms of shape functions evaluated at P, $\phi_i^k|_P$, and modal coordinates ${q^k}_i$. Finally, ν_k is the number of mode shapes associated with *Body k*.

$$\vec{r_p} = \vec{r_0} + \vec{\delta} \tag{13}$$

$$\vec{\delta} = \sum_{i=1}^{\nu_k} {\vec{\varphi_i}}^k q_i^k|_P \tag{14}$$

Let ${}^N V^{k+}$ represent the spatial velocity of handle J^{k+} with respect to the inertial reference frame N. The kinematic expressions for velocity are obtained as below.

$${}^N V^{k+} = \begin{bmatrix} {}^N\vec{\omega}^{k+} \\ {}^N\vec{v}^{k+} \end{bmatrix} \tag{15}$$

$${}^N V^P = \begin{bmatrix} {}^N\vec{\omega}^{p} \\ {}^N\vec{v}^{p} \end{bmatrix} = {}^N V^{k+} + \begin{bmatrix} \vec{0} \\ {}^N\vec{\omega}^{k+} \times (\vec{r_0} + \vec{\delta}) \end{bmatrix} + \Phi^k \dot{q}^k \tag{16}$$

$$= (S^{P/k+})^T \, {}^N V^{k+} + \Phi^k \dot{q}^k \tag{17}$$

$$\text{where} \quad \Phi^k \dot{q}^k = \sum_{i=1}^{\nu_k} \phi_i^k \dot{q}_i^k|_P \tag{18}$$

$$\text{with} \quad \phi_i^k = \begin{bmatrix} \vec{\psi}_i^k \\ \vec{\varphi}_i^k \end{bmatrix} \tag{19}$$

$$\text{and} \quad S^{P/k+} = \begin{bmatrix} \underline{U} & \vec{r_p}\times \\ \underline{0} & \underline{U} \end{bmatrix} \tag{20}$$

Similarly, let ${}^N A^{k+}$ represent the spatial acceleration of handle J^{k+} with respect to the inertial reference frame *N*. The kinematic expression for the spatial accelerations of arbitrary differential volume at P can be obtained as below with the understanding that summation is implied over the index *i* unless explicitly stated.

$${}^N A^{k+} = \begin{bmatrix} {}^N\vec{\alpha}^{k+} \\ {}^N\vec{a}^{k+} \end{bmatrix} \tag{21}$$

$${}^N A^P = (S^{P/k+})^T \, {}^N A^{k+} + {}^N {A_t}^{P/k+} + \Phi^k \ddot{q}^k \tag{22}$$

$$\text{where} \quad \Phi^k \ddot{q}^k = \sum_{i=1}^{\nu_k} {\phi_i}^k \ddot{q}_i^k|_P \tag{23}$$

$$\text{and} \quad {}^N {A_t}^{P/k+} = \begin{bmatrix} {}^N\vec{\omega}^{k+} \times {\psi_i}^k \dot{q}_i^k \\ {}^N\vec{\omega}^{k+} \times ({}^N\vec{\omega}^{k+} \times \vec{r_p}) + 2 {}^N\vec{\omega}^{k+} \times {\vec{\varphi}_i}^k \dot{q}_i^k \end{bmatrix}_P \tag{24}$$

Using relative coordinates, the angular and linear velocity of joint J^{k+} can be expressed in the local body basis in terms of the joint generalized speeds u and partial velocities ${}^N{\vec{\omega_r}}^{k+}$ and ${}^N{\vec{v_r}}^{k+}$ as

$${}^N\vec{\omega}^{k+} = {}^N\vec{\omega}^{k-} + {}^N{\vec{\omega_r}}^{k+} u_r \tag{25}$$

$${}^N\vec{v}^{k+} = {}^N\vec{v}^{k-} + {}^N{\vec{v_r}}^{k+} u_r \tag{26}$$

where summation is implied over the repeated index r.

From above equations, the partial velocities ${}^N\vec{v_r}^{\,k^+}$ and ${}^N\vec{\omega_r}^{\,k^+}$ for general spatial motion can be expressed as

$$
\begin{aligned}
P_r^{k+} &= \begin{bmatrix} {}^N\vec{\omega_r}^{\,k^+} \\ {}^N\vec{v_r}^{\,k^+} \end{bmatrix} && (27)\\
&= \begin{bmatrix} \hat{k}_i \\ 0 \end{bmatrix} \text{ for r} = \text{i},\ \text{i} = 1,2,3 && (28)\\
&= \begin{bmatrix} 0 \\ \hat{k}_i \end{bmatrix} \text{ for r} = \text{i}+3,\ \text{i} = 1,2,3 && (29)\\
&= \begin{bmatrix} 0 \\ 0 \end{bmatrix} \text{ for r} = \text{i}+6,\ \text{i} = 1,\cdots,\nu_{\text{k}} && (30)
\end{aligned}
$$

Similarly, the partial velocities ${}^N\vec{v_r}^{\,P}$ and ${}^N\vec{\omega_r}^{\,P}$ can be expressed as

$$
\begin{aligned}
P_r^P &= (S^{P/k^+})^T(P_r^{k+}) + \Phi_r^k|_P && (31)\\
&= \begin{bmatrix} \hat{k}_i \\ \hat{k}_i \times \vec{r_p} \end{bmatrix} \text{ for r} = \text{i},\ \text{i} = 1,2,3 && (32)\\
&= \begin{bmatrix} 0 \\ \hat{k}_i \end{bmatrix} \text{ for r} = \text{i}+3,\ \text{i} = 1,2,3 && (33)\\
&= \begin{bmatrix} \vec{\psi}_i^{\,k} \\ \vec{\varphi}_i^{\,k} \end{bmatrix}_P \text{ for r} = \text{i}+6,\ \text{i} = 1,\cdots,\nu_{\text{k}} && (34)
\end{aligned}
$$

Although in the above equations all the partial velocities for $r = 1 : \nu$ are shown, in the presence of a kinematic joint, corresponding terms that are constrained by the joint would be absent in the calculation of the partial velocities.

Another entity which is useful in the derivation of this algorithm is the map of the joint motion subspace and the orthogonal complement of this map. The matrix of joint motion subspace is exactly the partial velocity matrix associated with the rigid body degrees of freedom evaluated at the joint location. The active forces at the joint and the joint degrees of freedom lie in the space spanned by the column vector(s) of this matrix. It can be interpreted as the $6 \times dof$ matrix that maps the dof generalized speeds associated with the relative motions permitted by the joint into a 6×1 vector of spatial relative velocity across the joint. This matrix is hereafter referred to as $H^{k/(k+1)}$. Let $D^{k/(k+1)}$ be the orthogonal complement of the joint motion subspace matrix $H^{k/(k+1)}$. While $H^{k/(k+1)}$ is a $6 \times dof$ matrix corresponding to the $dof \times 1$ column of joint degrees of freedom, $D^{k/(k+1)}$ is a $6 \times (6 - dof)$ matrix that maps the directions imposed on the constrained degrees of freedom of the joint. For example, in a spherical joint, the translational degrees of motion are constrained while the rotational degrees of freedom are maintained. Hence the corresponding maps are given by

$$
H^{k/(k+1)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad D^{k/(k+1)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{35}
$$

By definition of the orthogonal complement $H^{k/(k+1)}$ and $D^{k/(k+1)}$ satisfy the following relation:

$$\left(H^{k/k+1}\right)^T \cdot D^{k/(k+1)} = \left(D^{k/k+1}\right)^T \cdot H^{k/(k+1)} = 0 \tag{36}$$

3.2 Flexible Body Kinetics

The equations of motion of a generic body of the system, *Body k*, can be written as below using a velocity projection formulation also known as Kane's [27] method.

$$\int_B {}^N\vec{v_r}^{\,P} \cdot ({}^N\vec{a}^{\,P}\rho\, dD) + {}^N\vec{\omega_r}^{\,P} \cdot \overbrace{({}^N\vec{\alpha}^{\,P}\, dI)}^{0} - [{}^N\vec{v_r}^{\,P} \cdot \vec{f}^{\,P} + {}^N\vec{v_r}^{\,k^+} \cdot \vec{f_c}^{\,k^+} + {}^N\vec{\omega_r}^{\,k^+} \cdot \vec{\tau_c}^{\,k^+}$$
$$+ {}^N\vec{v_r}^{\,k+1^-} \cdot \vec{f_c}^{\,k+1^-} + {}^N\vec{\omega_r}^{\,k+1^-} \cdot \vec{\tau_c}^{\,k+1^-}] \quad = 0 \tag{37}$$

Consider the constituent terms of the above equation separately. In the above equation $\int {}^N\vec{v_r}^{\,P} \cdot ({}^N\vec{a}^{\,P}\rho\, dD)$ is the Generalized Inertia Force term, where ${}^N\vec{a}^{\,P}$ can be expanded as below.

$$\begin{aligned}\int_{B_k} {}^N\vec{v_r}^{\,P} \cdot ({}^N\vec{a}^{\,P}\rho\, dD) \;=\; & \int_{B_k} {}^N\vec{v_r}^{\,P} \cdot [\,{}^N\vec{a}^{\,k^+} + {}^N\vec{\alpha}^{\,k^+} \times \vec{r_p} + {}^N\vec{\omega}^{\,k^+} \times ({}^N\vec{\omega}^{\,k^+} \times \vec{r_P}\,) \\ & + 2\,{}^N\vec{\omega}^{\,k^+} \times \vec{\varphi_i}^{\,k}\dot{q_i}^{\,k}|_P + \varphi_i{}^k\ddot{q_i}^{\,k}|_P\,]\rho\, dD\end{aligned} \tag{38}$$

For $r = i = 1, 2, 3$, associated with the spatial rotation of the *Body k* parent joint, the above becomes

$$\begin{aligned}\int_{B_k} {}^N\vec{v_r}^{\,P} \cdot ({}^N\vec{a}^{\,P}\rho\, dD) &= [\int \vec{r_p} \times (\hat{k_i} \times \vec{r_p})\rho dD)] \cdot {}^N\vec{\alpha}^{\,k^+} + \hat{k_i} \cdot [\int \vec{r_p}\rho dD \times {}^N\vec{a}^{\,k^+}] \\ -\hat{k_i} \cdot [{}^N\vec{\omega}^{\,k^+} \times \int (\vec{r_p}\vec{r_p}\rho dD) \cdot {}^N\vec{\omega}^{\,k^+}] &+ 2\,{}^N\vec{\omega}^{\,k^+} \cdot [\int \vec{\varphi_j}^{\,k}|_P \times (\hat{k_i} \times \vec{r_p})\rho dD]\dot{q_j}^{\,k} \\ &+ [\hat{k_i} \cdot \int (\vec{\varphi_j}^{\,k}|_P \times \vec{r_p}\rho dD)]\ddot{q_j}^{\,k}\end{aligned} \tag{39}$$

For $r = i + 3, (i = 1, 2, 3)$, associated with the spatial translation of the *Body k* parent joint, the expansion becomes

$$\begin{aligned}\int_{B_k} {}^N\vec{v_r}^{\,P} \cdot ({}^N\vec{a}^{\,P}\rho\, dD) &= \hat{k_i} \cdot ({}^N\vec{\alpha}^{\,k^+} \times \int \vec{r_p}\rho dD) + \hat{k_i} \cdot [{}^N\vec{\omega}^{\,k^+} \times ({}^N\vec{\omega}^{\,k^+} \times \int \vec{r_p}\rho dD)] \\ +\hat{k_i} \cdot \int (\rho\vec{\varphi_j}^{\,k}|_P dD)\ddot{q_j}^{\,k} &+ \hat{k_i} \cdot ({}^N\vec{a}^{\,k^+} \int \rho dD) + \hat{k_i} \cdot [2\,{}^N\vec{\omega}^{\,k^+} \times \int (\rho\vec{\varphi_j}^{\,k}|_P dD)]\dot{q_j}^{\,k}\end{aligned} \tag{40}$$

For $r = i + 6, (i = 1, \cdots, \nu_k)$, associated with the modal coordinates of *Body k*, it takes the form

$$\begin{aligned}\int_{B_k} {}^N\vec{v_r}^{\,P} \cdot ({}^N\vec{a}^{\,P}\rho\, dD) &= {}^N\vec{a}^{\,k^+} \int (\vec{\varphi_i}^{\,k}|_P \rho dD) + {}^N\vec{\alpha}^{\,k^+} \int (\vec{r_p} \times \vec{\varphi_i}^{\,k}|_P)\rho dD \\ + \int (\vec{\varphi_i}^{\,k}|_P \cdot \vec{\varphi_j}^{\,k}|_P \rho dD)\ddot{q_j}^{\,k} &+ {}^N\vec{\omega}^{\,k^+} \cdot \int ([\vec{r_p}\vec{\varphi_i}^{\,k}|_P - \vec{\varphi_i}^{\,k}|_P\vec{r_p}]\rho dD) \cdot {}^N\vec{\omega}^{\,k^+} \\ + \int [\vec{\varphi_i}^{\,k}|_P \cdot (2\,{}^N\vec{\omega}^{\,k^+} \times (\rho\vec{\varphi_j}^{\,k}|_P))]\dot{q_j}^{\,k} dD\end{aligned} \tag{41}$$

The above equations can be collected together and expressed in matrix format as

$$\int_B {}^N\vec{v_r}^P \cdot ({}^N\vec{a}^P \rho \, dD) = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} & \Gamma_{13} \\ \Gamma_{21} & \Gamma_{22} & \Gamma_{23} \\ \Gamma_{31} & \Gamma_{32} & \Gamma_{33} \end{bmatrix} \begin{bmatrix} \alpha^{k+} \\ a^{k+} \\ \ddot{q}^k \end{bmatrix} + \begin{bmatrix} \beta_{11} \\ \beta_{12} \\ \beta_{13} \end{bmatrix} \tag{42}$$

A closer inspection of above equations (39- 41) clearly identifies several terms in the mass matrix which retain the coupling between the finite rotations at the joints and the elastic deformation of the body. This clearly shows the mass matrix to be a nonlinear function of the joint and flexible body coordinates while the bias force is a nonlinear function of the coordinates and the speeds.

The equations above contain integrals over the volume of the body. These integrals (or summations in the case of discrete masses) produce time invariant coefficients for the temporally varying quantities *(q, $\dot{q}$, $\ddot{q}$)*. As such these coefficients need only be calculated once at the being (a pre-processing step) of the simulation.

From equation (37), the *Generalized Constraint Force* contribution from the joint J^{k+} is given by the $[{}^N\vec{v_r}^{k^+} \cdot \vec{f_c}^{k^+} + {}^N\vec{\omega_r}^{k^+} \cdot \vec{\tau_c}^{k^+}]$ term. For $r = i = 1, 2, 3$ it is expanded as

$$({}^N\vec{v_r}^{k^+} \cdot f_c^{\vec{k}+} + {}^N\vec{\omega_r}^{k^+} \cdot \tau_c^{\vec{k}+}) \quad = \hat{k_i} \cdot [(f_c^{\vec{k}+} \times \vec{r_p}) + \tau_c^{\vec{k}+}] = {\gamma_1}^{k^+} {F_c}^{k+1} \tag{43}$$

while for $r = i + 3, (i = 1, 2, 3)$ it is expressed as

$$({}^N\vec{v_r}^{k^+} \cdot f_c^{\vec{k}+} + {}^N\vec{\omega_r}^{k^+} \cdot \tau_c^{\vec{k}+}) \quad = \hat{k_i} \cdot f_c^{\vec{k}+} = {\gamma_2}^{k^+} {F_c}^{k+} \tag{44}$$

And for $r = i + 6, i = 1, \cdots, \nu_k$ it is expressed as

$$({}^N\vec{v_r}^{k^+} \cdot f_c^{\vec{k}+} + {}^N\vec{\omega_r}^{k^+} \cdot \tau_c^{\vec{k}+}) = [\vec{\varphi_i}^k|_{k+} \cdot f_c^{\vec{k}+} + \vec{\psi_i}^k|_{k+} \cdot \tau_c^{\vec{k}+}] = {\gamma_3}^{k^+} {F_c}^{k+} \tag{45}$$

In equation (37) $[{}^N\vec{v_r}^{k+1^-} \cdot \vec{f_c}^{k+1^-} + {}^N\vec{\omega_r}^{k+1^-} \cdot \vec{\tau_c}^{k+1^-}]$ is the *Generalized Constraint Force* term at joint J^{k+1}. It can be expanded for $r = i = 1, 2, 3$ as

$$({}^N\vec{v_r}^{k+1^-} \cdot \vec{f_c}^{k+1^-} + {}^N\vec{\omega_r}^{k+1^-} \cdot \vec{\tau_c}^{k+1^-}) \quad = \vec{f_c}^{k+1^-} \cdot (\hat{k_i} \times \vec{r}^{k+1^-}) + \hat{k_i} \cdot \vec{\tau_c}^{k+1^-} \tag{46}$$

$$= {\gamma_1}^{k+1^-} {F_c}^{k+1^-} \tag{47}$$

while for $r = i + 3, (i = 1, 2, 3)$ it is expressed as

$$({}^N\vec{v_r}^{k+1^-} \cdot \vec{f_c}^{k+1^-} + {}^N\vec{\omega_r}^{k+1^-} \cdot \vec{\tau_c}^{k+1^-}) = \hat{k_i} \cdot \vec{f_c}^{k+1^-} = {\gamma_2}^{k+1^-} {F_c}^{k+1^-} \tag{48}$$

And for $r = i + 6, i = 1, \cdots, \nu_k$ it is expressed as

$$({}^N\vec{v_r}^{k+1^-} \cdot \vec{f_c}^{k+1^-} + {}^N\vec{\omega_r}^{k+1^-} \cdot \vec{\tau_c}^{k+1^-}) \quad = [\vec{\varphi_i}^k|_{k+1^-} \cdot \vec{f_c}^{k+1^-} + \vec{\psi_i}^k|_{k+1^-} \cdot \vec{\tau_c}^{k+1^-}] \tag{49}$$

$$= {\gamma_3}^{k+1^-} {F_c}^{k+1^-} \tag{50}$$

$[{}^N\vec{v_r}^P \cdot \vec{f}^P]$ is the *Generalized Body Force* term. This term includes the stiffness terms originating from the deformation of the flexible body as well as gravitational and other body

forces. It can be expanded as

$$\text{for } r = i = 1, 2, 3$$
$$\int_{B_k} {}^N\vec{v_r}^P \cdot \vec{f^P} dD) = \hat{k_i} \cdot \int_B (\vec{f^P} \times \vec{r_p}) dD = \beta_{21} \tag{51}$$

$$\text{for } r = i + 3, (i = 1, 2, 3)$$
$$\int_{B_k} {}^N\vec{v_r}^P \cdot \vec{f^P} dD = \hat{k_i} \cdot \int_B \vec{f^P} dD = \beta_{22} \tag{52}$$

$$\text{for } r = i + 6, (i = 1, \cdots, \nu_k)$$
$$\int_{B_k} {}^N\vec{v_r}^P \cdot \vec{f^P} dD) = \int_B \vec{\varphi_i}^k|_P \cdot \vec{f^P} dD = \beta_{23} \tag{53}$$

Collecting all the above equations into a matrix form, the equations of motion for *Body k* can be expressed as

$$\begin{bmatrix} \Gamma_{11} & \Gamma_{12} & \Gamma_{13} \\ \Gamma_{21} & \Gamma_{22} & \Gamma_{23} \\ \Gamma_{31} & \Gamma_{32} & \Gamma_{33} \end{bmatrix}^k \begin{bmatrix} \alpha^{k+} \\ a^{k+} \\ \ddot{q}^k \end{bmatrix} - \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix}^{k+} F_c^{k^+} - \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix}^{k+1^-} F_c^{k+1^-} + \begin{bmatrix} \beta_{11} - \beta_{21} \\ \beta_{12} - \beta_{22} \\ \beta_{13} - \beta_{23} \end{bmatrix}^k = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{54}$$

The above matrix equations can be further consolidated in terms of the rigid body joint coordinates (variables) and the modal coordinates as

$$\begin{bmatrix} \Gamma_{RR} & \Gamma_{RF} \\ \Gamma_{FR} & \Gamma_{FF} \end{bmatrix}^k \begin{bmatrix} A \\ \ddot{q} \end{bmatrix}^{k+} - \begin{bmatrix} \gamma_R \\ \gamma_F \end{bmatrix}^{k^+} F_c^{k+} - \begin{bmatrix} \gamma_R \\ \gamma_F \end{bmatrix}^{k+1^-} F_c^{k+1^-} + \begin{bmatrix} \beta_R \\ \beta_F \end{bmatrix}^k = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{55}$$

Here the terms with subscripts *RR* and *R* are associated only with the rigid body joint degrees of freedom. The terms with subscript *FR* corresponds to the coupling between the rigid body joint degrees of freedom and the flexible body degrees of freedom. The terms with subscripts *FF* and *F* are associated solely with the flexible body degrees of freedom. The above are two sets of equations in terms of two sets of unknowns $\ddot{q}^k$ and A^{k+}. Solving the lower equation, an expression for $\ddot{q}^k$ can be obtained as shown below.

$$\Gamma_{FR}^k A^{k+} + \Gamma_{FF}^k \ddot{q}^k - \gamma_F^{k+} F_c^{k+} - \gamma_F^{k+1^-} F_c^{k+1^-} + \beta_F^k = 0 \tag{56}$$
$$\Rightarrow \ddot{q}^k = -{\Gamma_{FF}^k}^{-1} [\Gamma_{FR}^k A^{k+} - \gamma_F^{k+} F_c^{k+} - \gamma_F^{k+1^-} F_c^{k+1^-} + \beta_F^k] \tag{57}$$

Substituting the expression for $\ddot{q}^k$ in equation (56), an expression for A^k can be obtained as

$$[\Gamma_{RR}^k - \Gamma_{RF}^k {\Gamma_{FF}^k}^{-1} \Gamma_{FR}^k] A^{k+} - [\gamma_R^{k+} - \Gamma_{RF}^k {\Gamma_{FF}^k}^{-1} \gamma_F^{k+}] F_c^{k+} \\ -[\gamma_R^{k+1^-} - \Gamma_{RF}^k {\Gamma_{FF}^k}^{-1} \gamma_F^{k+1^-}] F_c^{k+1^-} + [\beta_R^k - \Gamma_{RF}^k {\Gamma_{FF}^k}^{-1} \beta_F^k] = 0 \tag{58}$$

$$\Rightarrow A^{k+} = {\zeta_{11}}^k F_c^{k+} + {\zeta_{12}}^k F_c^{k+1^-} + {\zeta_{13}}^k \tag{59}$$

The above expressions involve an inversion of the Γ_{FF}^k matrix. This matrix is local to $Body\ k$, is of dimension $\nu_k \times \nu_k$, and remains constant. It is diagonal if only orthogonal vibration modes are used. Adding static correction modes will introduce off-diagonal terms that produce coupling between the vibration and correction modes. The sparsity of this matrix is dependent on the

number of vibration and correction modes used. However, in all cases, the Γ^k_{FF} is a symmetric positive definite matrix and it remains constant through the simulation. The computational cost of inverting this matrix is thus a fixed cost in the pre-processing analysis and there is no repeated cost incurred during the simulation.

Now consider the spatial acceleration of joint J^{k+1^-} which is given by

$$A^{k+1^-} = (S^{k/k+1})^T A^{k+} + A_t{}^{k+1/k} + \Phi^{k+1/k^T} \ddot{q}^k \tag{60}$$

Substituting the expression for $\ddot{q}^k$ in the above equation, the equation becomes

$$\begin{aligned} A^{k+1^-} = (S^{k/k+1})^T A^{k+} + A_t{}^{k+1/k} - (\Phi^k|_{k+1^-})^T {\Gamma^k_{FF}}^{-1} [\Gamma^k_{FR} A^{k+} - \gamma_F^{k+} F_c^{k+} \\ -\gamma_F^{k+1^-} F_c^{k+1^-} + \beta_F^k] \end{aligned} \tag{61}$$

$$\begin{aligned} \Rightarrow A^{k+1^-} = [(S^{k/k+1})^T - (\Phi^k|_{k+1^-})^T {\Gamma^k_{FF}}^{-1} \Gamma^k_{FR}] A^{k+} + [(\Phi^k|_{k+1^-})^T {\Gamma^k_{FF}}^{-1} \gamma_F^{k+1^-}] F_c^{k+1^-} \\ +[(\Phi^k|_{k+1^-})^T {\Gamma^k_{FF}}^{-1} \gamma_F^k] F_c^{k+} + [A_t{}^{k+1/k} - (\Phi^k|_{k+1^-})^T {\Gamma^k_{FF}}^{-1} \beta_F^k] \end{aligned} \tag{62}$$

$$\Rightarrow A^{k+1^-} = \eta_1{}^k A^{k+} + \eta_2{}^k F_c^{k+} + \eta_3{}^k F_c^{k+1^-} + \eta_4{}^k \tag{63}$$

where η_i $(i = 1:4)$ simply represent useful intermediate coefficients which will aid in subsequent manipulations.

Now substituting the expression for A^k into equation (63), the expression for A^{k+1^-} can be obtained as

$$\begin{aligned} A^{k+1^-} &= [\eta_2{}^k + \eta_1{}^k \zeta_{11}{}^k] F_c^{k+} + [\eta_3{}^k + \eta_1{}^k \zeta_{12}{}^k] F_c^{k+1^-} + [\eta_4{}^k + \eta_1{}^k \zeta_{13}{}^k] && (64) \\ &= \zeta_{21}{}^k F_c^{k+} + \zeta_{22}{}^k F_c^{k+1^-} + \zeta_{23}{}^k && (65) \end{aligned}$$

Thus, from equations (59) and (65) the two handle articulated body equations for the flexible *body k* are given by

$$\begin{aligned} A^{k+} &= \zeta_{11}{}^k F_c^{k+} + \zeta_{12}{}^k F_c^{k+1^-} + \zeta_{13}{}^k && (66) \\ A^{k+1^-} &= \zeta_{21}{}^k F_c^{k+} + \zeta_{22}{}^k F_c^{k+1^-} + \zeta_{23}{}^k && (67) \end{aligned}$$

Similarly for *body (k+1)*, the two handle equations of motion are

$$\begin{aligned} A^{k+1^+} &= \zeta_{11}{}^{k+1} F_c^{k+1} + \zeta_{12}{}^{k+1} F_c^{k+2^-} + \zeta_{13}{}^{k+1} && (68) \\ A^{k+2^-} &= \zeta_{21}{}^{k+1} F_c^{k+1} + \zeta_{22}{}^{k+1} F_c^{k+2^-} + \zeta_{23}{}^{k+1} && (69) \end{aligned}$$

These equations are now in the same form as that of the two handle articulated body equations of motion found in Featherstone's Divide-and-Conquer algorithm for rigid bodies (RDCA) and as discussed in above section (2.1). These can now be coupled together to form the two handle equations of motion of the resulting assembly.

3.3 Recursive Expression for Inertia Coupling Terms

In the discussion of the general scheme of the Divide and Conquer Algorithm, it was mentioned that a set of recursive formulae is used to couple together the equations of motion of successive bodies in the system to form the equations of motion of the resulting assemblies. [18] derives a set of recursive formulae but recommends an alternate set of formulae (without derivation) for use in actual implementation for better computational efficiency. In this section, a derivation of this alternate set of formulae is presented.

The relative acceleration at the joint connecting *Body k* and *Body k+1* is given by the following equation.

$$ {}^{N}A^{k+1^{+}} - {}^{N}A^{k+1^{-}} = H^{k/(k+1)}\dot{u} + \dot{H}^{k/(k+1)}u \tag{70} $$

The constraint force on joint $J^{k+1^{+}}$ viz. $F_c^{k+1^{+}}$ and joint $J^{k+1^{-}}$ viz. $F_c^{k+1^{-}}$ are equal in magnitude and opposite in direction. This is a direct result of Newton's third law of motion. Using this fact and substituting the expressions for ${}^{N}A^{k+1^{-}}$ and ${}^{N}A^{k+1^{+}}$ from equations (67) and (68) into equation (70), an expression for $F_c^{k+1^{+}}$ is obtained as

$$ \begin{aligned} [{\zeta_{11}}^{k+1} + {\zeta_{22}}^{k}]F_c^{k+1^{+}} = [&{\zeta_{21}}^{k}F_c^{k+} - {\zeta_{12}}^{k+1}F_c^{k+2^{-}} + {\zeta_{23}}^{k} \\ &-{\zeta_{13}}^{k+1} + H^{k/(k+1)}\dot{u} + \dot{H}^{k/(k+1)}u] \end{aligned} \tag{71} $$

$$ \begin{aligned} \Rightarrow F_c^{k+1^{+}} = [{\zeta_{11}}^{k+1} + {\zeta_{22}}^{k}]^{-1}[&{\zeta_{21}}^{k}F_c^{k+} - {\zeta_{12}}^{k+1}F_c^{k+2^{-}} \\ &+{\zeta_{23}}^{k} - {\zeta_{13}}^{k+1} + H^{k/(k+1)}\dot{u} + \dot{H}^{k/(k+1)}u] \end{aligned} \tag{72} $$

Pre-multiplying equation (71) by $\left(D^{k/k+1}\right)^{T}$ gives

$$ \begin{aligned} (D^{k/k+1})^{T}[{\zeta_{11}}^{k+1} + {\zeta_{22}}^{k}]F_c^{k+1^{+}} = (D^{k/k+1})^{T}[&{\zeta_{21}}^{k}F_c^{k+} + {\zeta_{23}}^{k} - {\zeta_{13}}^{k+1} - {\zeta_{12}}^{k+1}F_c^{k+2^{-}} \\ &+\dot{H}^{k/(k+1)}u] + \underbrace{(D^{k/k+1})^{T}H^{k/(k+1)}}_{0}\dot{u} \end{aligned} \tag{73} $$

From the definition of the orthogonal complement of joint motion subspace, the constraint force $F_c^{k+1^{+}}$ can be expressed in terms of the measure numbers of the constraint torques ${\tilde{\tau}_c}^{k+1^{+}}$ and constraint forces ${\tilde{f}_c}^{k+1^{+}}$ as

$$ F_{MN}^{k+1^{+}} = \begin{bmatrix} {\tilde{\tau}_c}^{k+1^{+}} \\ {\tilde{f}_c}^{k+1^{+}} \end{bmatrix} \tag{74} $$

$$ F_c^{k+1^{+}} = D^{k/(k+1)}F_{MN}^{k+1^{+}} \tag{75} $$

Substituting the above into equation (73)

$$ \begin{aligned} (D^{k/k+1})^{T}[{\zeta_{11}}^{k+1} + {\zeta_{22}}^{k}]D^{k/(k+1)}F_{MN}^{k+1^{+}} = (D^{k/k+1})^{T}[&{\zeta_{21}}^{k}F_c^{k+} - {\zeta_{12}}^{k+1}F_c^{k+2^{-}} \\ &+{\zeta_{23}}^{k} - {\zeta_{13}}^{k+1} + \dot{H}^{k/(k+1)}u] \end{aligned} \tag{76} $$

The term $(D^{k/k+1})^{T}[{\zeta_{11}}^{k+1^{+}} + {\zeta_{22}}^{k}]D^{k/(k+1)}$ is a symmetric positive definite matrix and hence there is no problem associated with its inversion.

$$ \text{Let} \quad \mathrm{D^{k/(k+1)^{T}}}[{\zeta_{11}}^{k+1} + {\zeta_{22}}^{k}]\mathrm{D^{k/(k+1)}} = \widehat{\mathrm{X}} \tag{77} $$

$$ \Rightarrow F_{MN}^{k+1^{+}} = \widehat{X}^{-1}D^{k/(k+1)^{T}}[{\zeta_{21}}^{k}F_c^{k+} - {\zeta_{12}}^{k+1}F_c^{k+2^{-}} + {\zeta_{23}}^{k} - {\zeta_{13}}^{k+1} + \dot{H}^{k/(k+1)}u] \tag{78} $$

Pre-multiplying the above expression by $D^{k/(k+1)}$ to get the desired expression for $F_c^{k+1^{+}}$

$$ F_c^{k+1^{+}} = D^{k/(k+1)}F_{MN}^{k+1^{+}} \tag{79} $$

$$ = D^{k/(k+1)}\widehat{X}^{-1}D^{k/(k+1)^{T}}[{\zeta_{21}}^{k}F_c^{k+} - {\zeta_{12}}^{k+1}F_c^{k+2^{-}} + {\zeta_{23}}^{k} - {\zeta_{13}}^{k+1} + \dot{H}^{k/(k+1)}u] \tag{80} $$

The above expression for ${F_c^{k+1}}^+$ can be compactly written as below

$$
\begin{aligned}
{F_c^{k+1}}^+ &= \widehat{W}{\zeta_{21}}^k {F_c^k}^+ - \widehat{W}{\zeta_{12}}^{k+1} {F_c^k}^{2^-} + \widehat{Y} && (81)\\
\text{where} \quad \widehat{\mathrm{W}} &= D^{k/(k+1)} \widehat{X}^{-1} {D^{k/(k+1)}}^T && (82)\\
\text{and} \quad \widehat{\mathrm{Y}} &= \widehat{W}[{\zeta_{23}}^k - {\zeta_{13}}^{k+1} + \dot{H}^{k/(k+1)} u] && (83)
\end{aligned}
$$

The expression for ${F_c^{k+1}}^-$ is substituted in equation (66) and (69) and after some algebraic manipulation, the two handle equation of motion of the assembly of *Body k* and *Body (k+1)* are obtained as

$$
\begin{aligned}
A^{k+} &= [{\zeta_{11}}^k - {\zeta_{12}}^k \widehat{W} {\zeta_{21}}^k] {F_c^k}^+ + {\zeta_{12}}^k \widehat{W} {\zeta_{12}}^{k+1} {F_c^k}^{2^-} + {\zeta_{13}}^k - {\zeta_{12}}^k \widehat{Y} && (84)\\
A^{k+2^-} &= {\zeta_{21}}^{k+1} \widehat{W} {\zeta_{21}}^k {F_c^k}^+ + [{\zeta_{22}}^{k+1} - {\zeta_{21}}^{k+1} \widehat{W} {\zeta_{12}}^{k+1}] {F_c^k}^{2^-} + {\zeta_{23}}^{k+1} + {\zeta_{21}}^{k+1} \widehat{Y} && (85)
\end{aligned}
$$

From above, the recursive expression for Υ_{ij} can be obtained as

$$
\begin{aligned}
\Upsilon_{11} &= [{\zeta_{11}}^k - {\zeta_{12}}^k \widehat{W} {\zeta_{21}}^k] && (86)\\
\Upsilon_{22} &= [{\zeta_{22}}^{k+1} - {\zeta_{21}}^{k+1} \widehat{W} {\zeta_{12}}^{k+1}] && (87)\\
\Upsilon_{12} &= {\zeta_{12}}^k \widehat{W} {\zeta_{12}}^{k+1} && (88)\\
\Upsilon_{21} &= {\zeta_{21}}^{k+1} \widehat{W} {\zeta_{21}}^k && (89)\\
\Upsilon_{13} &= {\zeta_{13}}^k - {\zeta_{12}}^k \widehat{Y} && (90)\\
\Upsilon_{23} &= {\zeta_{23}}^{k+1} + {\zeta_{21}}^{k+1} \widehat{Y} && (91)
\end{aligned}
$$

These recursive formulae are used in the hierarchic assembly process to couple together the equations of motion of successive assemblies to form the two handle equations of motion of the resulting higher order assembly.

4 Computational Complexity

As indicated in section (2.1), each body of the system represents a single node at the lowest level of a binary tree. The algorithm works in four sweeps of the binary tree and the calculation of the spatial acceleration of the joints is carried out in $O(log(n_b))$ complexity (when performed in parallel) as explained in [18]. The traversal of the system topology in the binary tree form allows the processes to be time optimal $O(log(n_b))$ by using $O(n_b)$ processors. The binary tree is mapped directly onto $O(n_b)$ processors in a tree structure, where n_b in this context is the number of bodies (lowest level subassemblies) which make up the system. The architecture of the process is discussed in [18]-[19] and is not discussed further here.

Once the spatial accelerations of the handles on a constituent body have been determined, the time derivatives of the modal coordinates for the body are uncoupled and can be calculated independent of the other bodies of the system. As indicated in equation (57), the expression for the time derivatives of the modal coordinates involves a matrix inversion of the term Γ_{FF}^k. If the admissible mode shapes for a body comprise only of natural modes of vibration which are mutually orthogonal, this matrix is diagonal. Hence the cost associated with its inversion is $O(\nu_i^3)$. However if the admissible mode shapes chosen are not orthogonal, as in the case of static correction modes, the matrix is no longer diagonal. Depending on the number of non-orthogonal mode shapes, the matrix is sparsely populated with the presence of off-diagonal terms which represent the coupling between non-orthogonal modes. In the worst case where there is coupling between all modes chosen for the body, the matrix is fully populated and

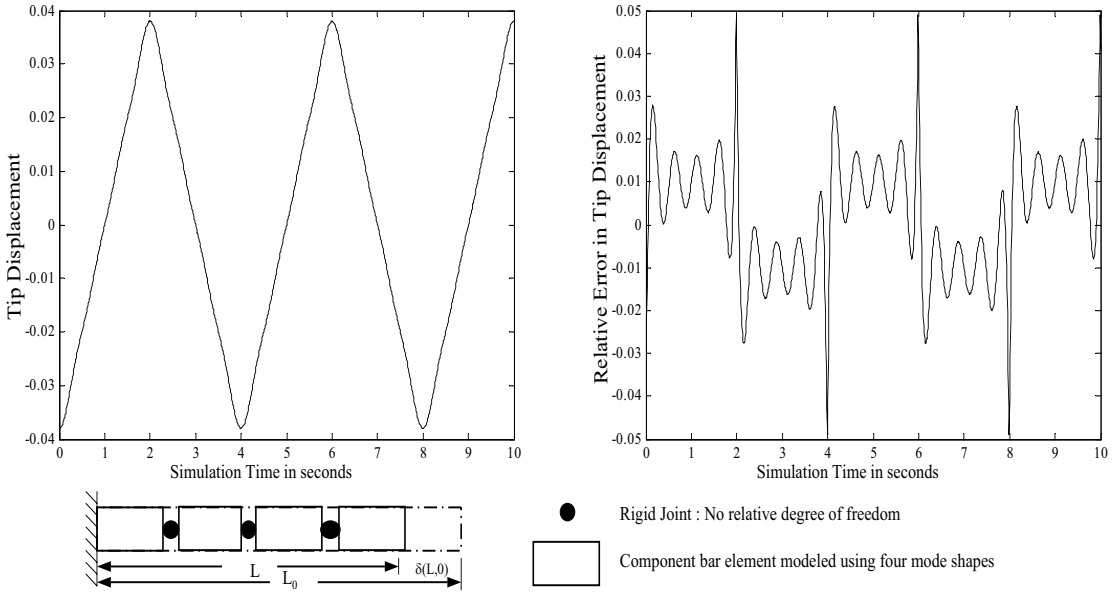

Figure 4: Longitudinal Vibration of Substructured Bar

the cost associated with its inversion is $O(max(\nu)^3)$. However, the matrix Γ^k_{FF} contains time invariant coefficients of the modal degrees of freedom, and hence remains constant during the simulation. Thus its inversion need only be calculated once in a pre-processing step, and as such this cost should not be considered in the cost per temporal integration step determination. The cost which is incurred at each evaluation is the $\Gamma_{RF}\Gamma^{-1}_{FF}$ multiplication which appears in equation (58). This is because matrix Γ_{RF} is state dependent and will thus vary with each step of temporal integration. For a fully populated Γ^k_{FF}, this matrix multiplication yields a maximum cost of $O(\nu_i)^2$ per integration step.

In the binary tree mapping, each body is mapped onto an individual processor. Hence the matrix multiplication $\Gamma_{RF}\Gamma^{-1}_{FF}$ can be independently calculated in $O(\nu_i)^2$ complexity on each processor unless some additional parallel methods are implemented to specifically deal with the possibly large matrix multiplications. These will not be discussed here. Instead, the performance of the method will be crudely determined by assuming that all flexible body manipulations associated with *Body k* are restricted to the single processor to which *Body k* is mapped. The maximum complexity of this process is thus $O(max(\nu_i)^2)$, where $max\ (\nu_i)$ is the maximum number of assumed modes on any constituent body of the system. The effective computational cost (which manifests itself as wall time) of the algorithm can thus be calculated as $O(log(n_b)) + O(max(\nu_i)^2)$ when implemented in parallel on processor optimal $O(n_b)$ processors.

5 Numerical Examples

To validate the algorithm presented here, simulation results from the modelling of two test cases are presented here. The first test case is the modelling of the longitudinal vibrations of an uniform bar. The bar is modelled as an elastic body, clamped at one end and free at the other. The modulus of elasticity and mass density of the rod is assumed to be unity. The length of the

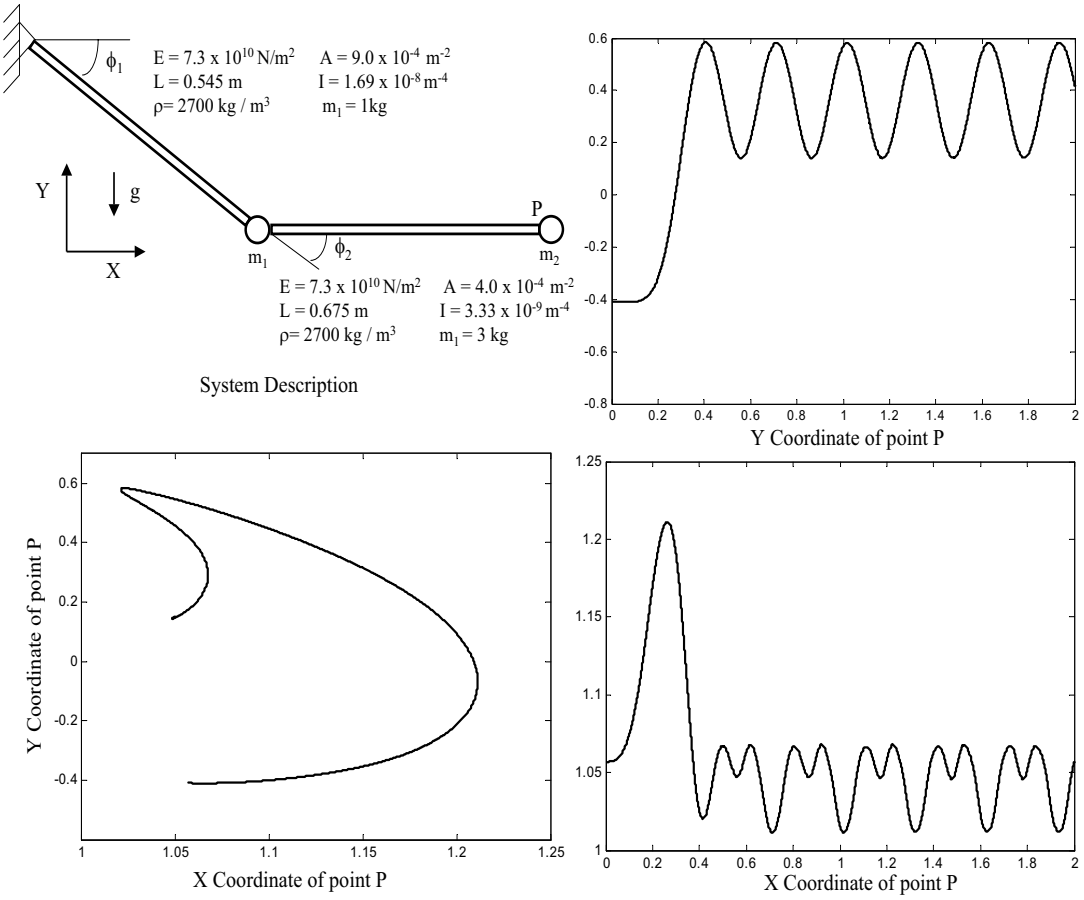

Figure 5: Two Bar Example

rod is 4 units and it is released from rest under an initial state of compressive strain of 0.01. The bar is sub-structured into four bar elements, each of unit length. Each bar element is connected to its parent element by rigid joints, i.e. no rigid body degrees of freedom are maintained at the joint between two consecutive bodies. The flexibility of the bar elements is modelled using assumed modes with the first four natural modes of vibration of a clamped-free bar chosen as the admissible shape functions. This system was chosen because the system behavior is easy to visualize and the exact analytical solution is readily available. The result shown in figure (4) is for a 10 second simulation and presents the tip displacement, as well as the displacement error (difference between the analytic solution and the FDCA solution) time histories. The result is in good agreement with the analytical solution, with error of form and magnitude appropriate for the approximation of this continuous system. Since a truncated set of modes are used, the model fails to capture the exact behavior at the tip. This is a characteristic behavior of the assumed mode modelling technique and not a shortcoming of the algorithm. The implementation was carried out in Matlab™ and the Matlab™ $ode45$ was used for numerical integration.

The second example is associated with the articulate flexible two member arm system shown in figure (5). The system consists of two elastic bars, each with point masses at the end. The two joints in the system are revolute and the angular motions of the two bars are prescribed as below. The transverse and longitudinal vibrations of each arm are modelled using two shape functions,

one for each vibration mode as shown in equations (94-95). This system has been simulated in [28]-[29]-[30] and established results are available for this problem. The system requires appropriate handling of the geometric stiffening effect and in the implementation here, the modelling approach outlined in [31]-[32] is used. The system is started from static equilibrium and undergoes prescribed angular motion for 0.5 seconds. The prescribed motion of the angles are shown in equations (92-93). After that, the system performs harmonic oscillations under the effect of gravity and internal strain energy. The results show the variation of the position of the tip (point P) with time. The results presented here are in agreement with the solutions in [28]-[29]-[30]. The implementation of the algorithm is clearly able to handle the geometric stiffening effect and accurately capture the dynamics of the system. This implementation too was carried out in Matlab™ and the numerical integration carried out using the Matlab™ $ode45$.

$$\begin{aligned}
\phi_1 &= -\pi/4 \cdots -\infty < t < 0 && (92)\\
&= \pi/4(-1+72t^3) \cdots 0 < t < 1/6\\
&= \pi/4(-18t+108t^2-144t^3) \cdots 1/6 \leq t < 1/3\\
&= \pi/4(-8+54t-108t^2+72t^3) \cdots 1/3 \leq t < 1/2\\
&= \pi/4 \cdots t > 1/2\\
\phi_2 &= -\phi_1 && (93)\\
\text{Longitudinal Vibration} &: (\frac{x}{L})^2 && (94)\\
\text{Transverse Vibration} &: 1.5(\frac{x}{L})^2 - 0.5(\frac{x}{L})^3 && (95)
\end{aligned}$$

6 Conclusion and Future Work

A new algorithm for solving the equations of motion of articulated flexible body systems is presented in this paper. The equations of motion for an unconstrained articulated body system comprising of arbitrary number of flexible and rigid bodies connected together by kinematic joints can be efficiently generated and solved using this algorithm. The computational complexity of this algorithm when implemented on $O(n_b)$ processors is expected to be $O(log(n_b)) + O(max(\nu_i)^2)$, where max ν_i is the maximum number of admissible shape functions for any constituent body of the system. The algorithm follows a divide and conquer scheme similar to the one presented in [18]. The elastic deformation of the component bodies is modelled by using superposition of a truncated set of admissible shape functions. Other than the use of superposition of a truncated set of component modes, no approximations are made. If the elastic deformations in the algorithm are neglected, the algorithm reduces to an exact algorithm for rigid body articulated systems. Although the equations derived above are for chain systems, the extension to tree topologies is trivial. [19] contains a detailed discussion on the accuracy of the RDCA and suggests a pivoting scheme for improving accuracy in a rigid body context. The discussions in [19] are generic to the DCA scheme and it is expected that they would be applicable to the present work. A detailed analysis and comparison of the numerical accuracy of the algorithm, as well as the implementation of this algorithm for systems with kinematically closed loops are discussed in a forthcoming paper.

Acknowledgement

This work was funded by the NSF NIRT Grant Number 0303902. The authors would like to thank the funding agency for their support.

REFERENCES

[1] Hooker W. W. and Margulies G., 1965. "The dynamical attitude equations for an n-body satellite". Journal of the Astronautical Sciences, 7(4), pp. 123:128, Winter.

[2] Walker M. W. and Orin D. E., 1982. "Efficient dynamic computer simulation of robotic mechanisms". Journal of Dynamic Systems, Measurement, and Control, 104, pp. 205-211, Sep.

[3] Armstrong W. W., 1979. "Recursive Solution to the Equations of Motion of an N-Link Manipulator". Fifth World Congress on the Theory of Machines and Mechanisms, **2**, pp. 1342–1346.

[4] Featherstone R. 1983 "The calculation of robotic dynamics using articulated body inertias". International Journal of Robotics Research, 2(1), pp. 13:30, Spring.

[5] Brandl H., Johanni R. and Otter M., 1986. "A Very Efficient Algorithm for the Simulation of Robots and Similar Multibody Systems Without Inversion of the Mass Matrix". IFAC/IFIP/IMACS Symposium, Vienna, Austria, pp. 95–100.

[6] Bae D. S., and Haug E. J. 1987. "A recursive formation for constrained mechanical system dynamics: Part I, Open loop systems". Mechanisms, Structures, and Machines, 15(3), pp. 359:382.

[7] Kurdila A. J., Menon R. G. and Sunkel J. W. 1993. "Nonrecursive Order N Formulation of Multibody Dynamics". Journal of Guidance, Control, and Dynamics, **16**, (5), pp. 838–844, Sept–Oct.

[8] Anderson K. S., 1995. "An Order-N formulation for the motion simulation of general multi-rigid-body tree system". Computers and Structures, **46** (3), pp. 547–559.

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

[10] Kim S. S. and Haug E. J. 1988. "Recursive Formulation for Flexible Multibody Dynamics: Part I, Open-Loop Systems". Computer Methods in Applied Mechanics and Engineering, **71**, (3), pp. 293–314.

[11] Jain A. K., Rodriguez G., 1992. "Recursive flexible multibody system dynamics using spatial operators". Journal of Guidance, Control and Dynamics, **15** (6), pp. 1453–1466.

[12] Anderson K. S., 1995. "Efficient Modelling of General Multibody Dynamic Systems with Flexible Components". Computational Dynamics in Multibody Systems, (), pp. 79–97. Kluwer Academic Publishers, Printed in Netherlands.

[13] Bauchau O. A., 2003. "Formulation of Modal-Based Elements in Nonlinear, Flexible Multibody Dynamics". International Journal of Multiscale Computational Engineering, **1** (2-3), pp. 161–180.

[14] Banerjee A. K., 1993. "Block-diagonal equations for multibody elastodynamics with geometric stiffness and constraints". Journal of Guidance, Control and Dynamics, **16** (6), pp. 1092–1100.

[15] Shabana A. A., 1997. "Flexible Multibody Dynamics: Review of Past and Recent Developments". Multibody System Dynamics, **1**, pp. 189–222.

[16] Quinn M., 1994. *Parallel Computing, Theory and Practice*. McGraw-Hill, New York.

[17] Fijany A., Shraf I. D'Eleuterio G. M. T. 1995. "Parallel O(log N) computation of manipulator forward dynamics". IEEE Trans. Robotics and Automation, **11** (3), pp. 389–400.

[18] Featherstone R., 1999. "A Divide-and-Conquer Articulated-Body Algorithm for Parallel O(log(n)) Calculation of Rigid Body Dynamics. Part 1: Basic Algorithm". The International Journal of Robotic Research, **18** (3), pp. 867–875.

[19] Featherstone R., 1999. "A Divide-and-Conquer Articulated-Body Algorithm for Parallel O(log(n)) Calculation of Rigid Body Dynamics. Part 2: Trees, Loops and Accuracy". The International Journal of Robotic Research, **18** (3), pp. 876–892.

[20] Shabana A. A., 1996. "Resonance Conditions and Deformable Body Coordinate Systems". Journal of Sound and Vibration, **192** (1), pp. 389–398.

[21] Schwertassek R., Wallrapp O. and Shabana A. 1999. "Flexible Multibody Simulation and Choice of Shape Functions". Nonlinear Dynamics, **20**, pp. 361–380.

[22] Schwertassek R., Dombrowski S. and Wallrapp O. 1999. "Modal Representation of Stree in Flexible Multibody Simulations". Nonlinear Dynamics, **20**, pp. 381–399.

[23] Shabana A. A., 1993. "Substructuring in Flexible Multibody Dynamics". Proceedings of Computer Aided Analysis of Rigid and Flexible Mechanical Systems: NATO-Advanced Study Institute, **1**, pp. 377–396.

[24] Kim S. S., and VanderPloeg M. J., 1986. "Generalized and Efficient Method for Dynamic Analysis of Mechanical Systems Using Velocity Transforms". Journal of Mechanism, Transmissions, and Automation Design, **108** (2), pp. 176–182.

[25] Nikravesh P. E., 1990. "Systematic Reduction of Multibody Equations to a Minimal Set". International Journal of Non-Linear Mechanics, **25** (2-3), pp. 143–151.

[26] Ambrosia J. A. C., 2001. "Complex Flexible Multibody Systems with Application to Vehicle Dynamics". Multibody System Dynamics **6** (2), pp. 163–182.

[27] Kane T. R. and Levinson D. A., 1985. *Dynamics: Theory and Application*. Mcgraw-Hill, New York.

[28] Botz M. and Hagedorn P., 1997. "Dynamic Simulation of multibody systems including planar elastic beams using Autolev". Engineering Computations, **14** (4), pp. 456–470.

[29] Claus H., 2001. "A Deformation Approach to Stress Distribution in Flexible Multibody Systems". Multibody System Dynamics, **6**, pp. 143–161.

[30] Seo S. and Yoo H., 2002. "Dynamic Analysis of Flexible Beams Undergoing Overall Motion Employing Linear Strain Measures". AIAA Journal, **40** (2), pp. 319–326.

[31] Yoo H., Ryan R. and Scott R., 1995. "Dynamics of Flexible Beams Undergoing Overall Motion". Journal of Sound and Vibration, **181** (2), pp. 261–278.

[32] Yoo H., and Shin S., 1998. "Vibration Analysis of Rotating Cantilever Beams". Journal of Sound and Vibration, **212** (5), pp. 807–828.

[33] Featherstone R., 1987. *Robot Dynamics Algorithms*. Kluwer Academic Publishers, New York.

[34] Shabana A. A., 1998. *Dynamics of Multibody Systems - Second Edition*. Cambridge University Press, New York.

[35] Anderson K. S., 1990. *Recursive Derivation of Explicit Equations of Motion for Efficient Dynamic/Control Simulation of Large Multibody Systems*. PhD Thesis, Stanford University, Stanford, CA, May.