A quasicoordinate formulation for dynamic simulation of complex multibody systems with constraints

G. Nelson, R. Quinn
Department of Mechanical and Aerospace Engineering, Case Western Reserve University, Cleveland, Ohio, USA

Abstract

A new extension of Lagrange's equation in terms of quasicoordinates is expanded to introduce a method of simulating complex multibody systems with many constraints. The extension uses a "global" form of velocity to permit a single matrix equation to be formulated, in a form similar to Newton's Second Law, which completely describes the motion of each "dynamic limb" of the system. Through a series of transformations the equations of motion can be expressed in an arbitrary independent set of generalized coordinates. The same coordinate transformations permit the incorporation of system constraints, such as clutches, brakes, transmissions, and coupling. The nature of the resulting algorithm is such that constraints can be changed easily during simulation runtime as desired by the user. As an example, a model and simulation of a unique prototype walking robot is described. The robot uses 17 revolute joints which are actuated through a clutching system and are powered by 5 independent motors.

1 Introduction

Much work has been done on the problem of modeling multibody dynamic systems. In the 1960's a formalism was developed for systems of rigid bodies forming a "topological tree" (Hooker and Margulies[1], and Roberson and Wittenburg[2]). In the 1970's software packages were developed for analysis of multibody systems: NBOD by Frisch[3] and DISCOS by Bodley et al.[4]. Several other software-packages have been developed since then for this problem, some including the treatment of flexible bodies. Books have been published on the topic (Roberson and Schwertassek[5] and Amirouche[6]).

"Order-n" recursive algorithms have been developed for modeling open kinematic chains of bodies (Brandl et al.[7]). An efficient algorithm for dynamic simulation of simple closed-chain mechanisms was recently developed by Lilly and Orin[8]. Computationally efficient algorithms are useful for real-time simulation and for model-based control methods. However, the tradeoffs are that the solution is implicit and some types of kinematic constraints may not be applicable.

Lagrange's equation in terms of quasicoordinates (Meirovitch[9]) is especially useful for formulating the equations of motions (EOM) for structures which undergo finite rotations in three dimensions. In previous work, Quinn[10] developed a formulation of Lagrange's equation in terms of quasicoordinates which is useful for developing the EOM for structures where the kinetic and potential energies are functions of angular velocity and orientation. Quinn and Chang[11] used this approach to formulate EOM for a chain of bodies in terms of absolute (inertial) quasicoordinates where each body had three revolute DOF. They then transformed these EOM into a form in terms of joint coordinates with each joint constrained to one, two or three DOF. Based on this, we developed a new, more compact representation of the dynamics of a multibody chain with revolute joints and a translating and rotating base frame (Nelson[12], Nelson and Quinn[13]). This formulation utilizes a "global" form of velocity to express the kinetic energy in terms of absolute angular velocities and orientations. In conjunction with several identities, this formulation leads to a single matrix equation, in a form similar to Newton's Second Law, which represents the entire set of rotational equations of motion for a system. The same "global" expressions produce compact forms of the translational EOM, and also lead to a simple representation of the effects of gravitational potential. To demonstrate the use of these equations, a simulation of a *Blaberus discoidalis* cockroach was developed. The simulated cockroach consisted of a 36 degree of freedom (DOF) model. Six 5 DOF legs supported a freely translating and rotating body. Results compared favorably with experimental observations.

The Lagrangian quasicoordinate formulation has two powerful features. Firstly, because the kinetic energy of a system can be expressed compactly in terms of quasicoordinates, it leads to an efficient form of the EOM wherein Coriolis terms can be eliminated. The second is that the these absolute quasicoordinate EOM (AQEOM) can be readily transformed to EOM in terms of a set of independent generalized coordinates. In previous work, the AQEOM were transformed to EOM in terms of joint coordinates (Quinn and Chang[11], Nelson and Quinn[13]).

In this paper, a method of constructing the EOM for uniquely constrained tree-like multibody systems is described. The AQEOM form a core "engine" which describe the dynamics of the system in terms of absolute angular velocities. We show that the AQEOM can be transformed to EOM in terms of any set of independent generalized coordinates. These same coordinate transformations permit the inclusion of many types of constraints on the system (e.g. a set of joints may be coupled). Also, since the AQEOM remain unchanged, changing the system constraints remains peripheral to the core dynamics of the system. This, naturally, leads to a simple algorithm for including, changing, and removing constraints while dynamic simulation is underway (e.g. joints may be alternately coupled and uncoupled to model clutching and braking).

An example use of this type of algorithm is presented. A prototype walking robot has been designed with the aid of this simulation method. The robot is being designed and manufactured by K^2T, Inc. in Pittsburgh, Pennsylvania, USA. The robot has 17 revolute joints, consisting of eight 2 DOF legs, and one body articulation. The robot is to be driven using 5 independent motors. A clutching and transmission system will both drive and lock joints according to commands from a behavior control and the current locomotion scenario. Therefore, the modeling and simulation of this robot, including the various constraints involved, provides an excellent example for our purposes.

2 EOM for a Multibody System in Absolute Quasicoordinates

As a review, consider a multibody system, ℓ, described as a chain of n rigid bodies connected with revolute joints. We shall refer to this chain of bodies as a 'limb'. This system is under the influence of a constant gravitational acceleration, $\underset{\sim}{g}$, and also contains a base frame fixed in the '1', or first, body, which is free to translate and rotate in three dimensional space. It can be shown that for any such system, the AQEOM can be expressed as (Nelson and Quinn[13])

$$\left[\begin{array}{c|c} I_v^T \bar{m}_\ell I_v & I_v^T \bar{m}_\ell \Phi_\ell \\ \hline \Phi_\ell^T \bar{m}_\ell I_v & \Phi_\ell^T \bar{m}_\ell \Phi_\ell + I_\ell^* \end{array}\right] \left[\begin{array}{c} \underset{\sim}{\ddot{R}}_{1\ell} \\ \hline \underset{\sim}{\dot{\omega}}_\ell \end{array}\right] + \left[\begin{array}{c} I_v^T \bar{m}_\ell \dot{\Phi}_\ell \\ \hline \Phi_\ell^T \bar{m}_\ell \dot{\Phi}_\ell + \tilde{\omega}_\ell I_\ell^* \end{array}\right] \underset{\sim}{\omega}_\ell - \left[\begin{array}{c} I_v^T \bar{m}_\ell I_v \\ \hline \Phi_\ell^T \bar{m}_\ell I_v \end{array}\right] \underset{\sim}{g} = \left[\begin{array}{c} \underset{\sim}{F}_\ell \\ \hline \underset{\sim}{M}_\ell \end{array}\right], \quad (1)$$

where the following definitions apply:

$$I_v^T = \left[I \;\vdots\; I \;\vdots\; \cdots \;\vdots\; I \right]^T_{(3\mathrm{x}3n)}, \qquad I = \text{(3x3) identity matrix}, \quad (2)$$

$$\bar{m}_\ell = \begin{bmatrix} \bar{m}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \bar{m}_n \end{bmatrix}_{\ell,(3n\mathrm{x}3n)}, \qquad \bar{m}_{i\ell} = \begin{bmatrix} m_i & 0 & 0 \\ 0 & m_i & 0 \\ 0 & 0 & m_i \end{bmatrix}_{\ell,(3\mathrm{x}3)}, \quad (3)$$

and m_i is the mass of body i. $\underset{\sim}{R}_{1\ell}$ is the position of the base frame of the system, expressed with respect to an inertial reference frame (N-frame),

$$\underset{\sim}{R}_{1\ell}^T = \left[R_x \quad R_y \quad R_z \right]_{\ell,(1\mathrm{x}3)}. \quad (4)$$

$\underset{\sim}{\omega}_\ell$ is a column vector composed of the absolute angular velocities, $\underset{\sim}{\omega}_i$, of the separate rigid bodies throughout the limb ℓ,

$$\underset{\sim}{\omega}_\ell^T = \left[\underset{\sim}{\omega}_1^T \;\vdots\; \underset{\sim}{\omega}_2^T \;\vdots\; \cdots \;\vdots\; \underset{\sim}{\omega}_n^T \right]_{\ell,(1\mathrm{x}3n)}, \quad (5)$$

with $\underset{\sim}{\omega}_i$ being expressed with respect to the i-frame which is attached to body i. Therefore,

$$\tilde{\omega}_\ell = \begin{bmatrix} \tilde{\omega}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \tilde{\omega}_n \end{bmatrix}_{\ell,(3n\times 3n)}, \qquad \tilde{\omega}_i = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}_{i,(3\times 3)}, \tag{6}$$

where $\tilde{\omega}_i$ is a skew symmetric matrix operation on the vector $\underset{\sim}{\omega}_i$. The Φ_ℓ matrix comes from a global representation of n (3x1) velocity vectors corresponding to the n bodies in ℓ,

$$\Phi_\ell = \begin{bmatrix} C_{N1}\tilde{S}_1^T & 0 & \cdots & 0 \\ C_{N1}\tilde{L}_1^T & C_{N2}\tilde{S}_2^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C_{N1}\tilde{L}_1^T & C_{N2}\tilde{L}_2^T & \cdots & C_{Nn}\tilde{S}_n^T \end{bmatrix}_{\ell,(3n\times 3n)}, \tag{7}$$

where C_{Ni} is an orthogonal rotational transformation matrix from the i-frame to the N-frame, $\underset{\sim}{L}_i$ is the position of the (i+1)-frame origin in the i-frame, expressed with respect to the i-frame, and $\underset{\sim}{S}_i$ is the position of the center of mass of body i, expressed with respect to the i-frame. The matrix $\dot{\Phi}_\ell$ can be found by taking the time derivative of Φ_ℓ and recalling that

$$\dot{C}_{Ni} = C_{Ni}\tilde{\omega}_i . \tag{8}$$

Furthermore,

$$I_\ell^* = \begin{bmatrix} I_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & I_n \end{bmatrix}_{\ell,(3n\times 3n)}, \tag{9}$$

where I_i is the rotational inertia matrix of body i about its center of mass, measured with respect to the i-frame. Finally, $\underset{\sim}{F}_\ell$ represents the net force acting on the base frame of the limb, and $\underset{\sim}{M}_\ell$ is a (3nx1) vector comprised of the applied absolute moments acting on each body.

3 Transforming the EOM to True Coordinates

In general, the rotational EOM for the limb ℓ can be expressed meaningfully in terms of three different types of coordinates: absolute quasicoordinates, $\underset{\sim}{\omega}_\ell$, with applied moments $\underset{\sim}{M}_\ell$, relative quasicoordinates, $\underset{\sim}{\Omega}_\ell$, with applied moments $\underset{\sim}{M}_{rel,\ell}$, and true coordinates, $\underset{\sim}{\alpha}_\ell$, with applied torques $\underset{\sim}{\tau}_\ell$, the last being analogous to actuated DOFs. The following relationship exists between the two types of quasicoordinates:

$$\underset{\sim}{\omega}_\ell = C_\ell^R \underset{\sim}{\Omega}_\ell, \tag{10}$$

where

$$\mathrm{C}_\ell^{\mathrm{R}} = \begin{bmatrix} I & 0 & \cdots & 0 & 0 \\ \mathrm{C}_{21} & I & \cdots & 0 & 0 \\ \mathrm{C}_{31} & \mathrm{C}_{32} & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & I & 0 \\ \mathrm{C}_{n1} & \mathrm{C}_{n2} & \cdots & \mathrm{C}_{n(n-1)} & I \end{bmatrix}_{\ell,(3n\times 3n)} \tag{11}$$

Applying the principle of virtual work then provides the following relationship:

$$\underset{\sim}{\mathrm{M}}_{\mathrm{rel},\ell} = \mathrm{C}_\ell^{\mathrm{RT}} \underset{\sim}{\mathrm{M}}_\ell . \tag{12}$$

Further, the relative quasicoordinates can be related to the true coordinates as follows:

$$\underset{\sim}{\Omega}_\ell = \mathrm{D}_\ell \dot{\underset{\sim}{\alpha}}_\ell , \tag{13}$$

where

$$\mathrm{D}_\ell = \begin{bmatrix} \mathrm{D}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathrm{D}_n \end{bmatrix}_{\ell,(3n\times k)} , \tag{14}$$

where k is the number of independent rotational DOFs of the limb. Virtual work leads to the following relationship:

$$\underset{\sim}{\tau}_\ell = \mathrm{D}_\ell^{\mathrm{T}} \underset{\sim}{\mathrm{M}}_{\mathrm{rel},\ell} . \tag{15}$$

Eqns.(12) and (15) provide a means by which the AQEOM can be transformed into true coordinate EOM:

$$\begin{aligned} (\text{ true coordinate EOM})_\ell &= \mathrm{D}_\ell^{\mathrm{T}}(\text{ relative quasicoordinate EOM})_\ell \\ &= \mathrm{D}_\ell^{\mathrm{T}}\mathrm{C}_\ell^{\mathrm{RT}}(\text{ AQEOM})_\ell . \end{aligned} \tag{16}$$

We now have everything needed to present a final form for the EOM of the limb ℓ in terms of true coordinates:

$$\overline{\mathrm{D}}_\ell = \begin{bmatrix} I & 0 \\ 0 & \mathrm{D}_\ell \end{bmatrix}, \qquad \overline{\mathrm{C}}_\ell^{\mathrm{R}} = \begin{bmatrix} I & 0 \\ 0 & \mathrm{C}_\ell^{\mathrm{R}} \end{bmatrix}, \tag{17}$$

$$\left[\overline{\mathrm{D}}_\ell^{\mathrm{T}} \overline{\mathrm{C}}_\ell^{\mathrm{RT}} \begin{bmatrix} I_v^{\mathrm{T}} \overline{\mathrm{m}}_\ell I_v & I_v^{\mathrm{T}} \overline{\mathrm{m}}_\ell \Phi_\ell \\ \Phi_\ell^{\mathrm{T}} \overline{\mathrm{m}}_\ell I_v & \Phi_\ell^{\mathrm{T}} \overline{\mathrm{m}}_\ell \Phi_\ell + \mathrm{I}_\ell^{*} \end{bmatrix} \overline{\mathrm{C}}_\ell^{\mathrm{R}} \overline{\mathrm{D}}_\ell \right] \begin{bmatrix} \ddot{\underset{\sim}{\mathrm{R}}}_{1\ell} \\ \ddot{\underset{\sim}{\alpha}}_\ell \end{bmatrix} = \begin{bmatrix} \overline{\underset{\sim}{\mathrm{F}}}_\ell \\ \overline{\underset{\sim}{\tau}}_\ell \end{bmatrix}, \tag{18}$$

where

$$\begin{bmatrix} \overline{\underset{\sim}{\mathrm{F}}}_\ell \\ \overline{\underset{\sim}{\tau}}_\ell \end{bmatrix} = -\overline{\mathrm{D}}_\ell^{\mathrm{T}} \overline{\mathrm{C}}_\ell^{\mathrm{RT}} \begin{bmatrix} I_v^{\mathrm{T}} \overline{\mathrm{m}}_\ell \underset{\sim}{\Gamma}_\ell \\ \Phi_\ell^{\mathrm{T}} \overline{\mathrm{m}}_\ell \underset{\sim}{\Gamma}_\ell + \widetilde{\omega}_\ell \mathrm{I}_\ell^{*} \underset{\sim}{\omega}_\ell + \mathrm{I}_\ell^{*} \mathrm{C}_\ell^{\mathrm{R}} \underset{\sim}{\xi}_\ell \end{bmatrix} + \begin{bmatrix} \underset{\sim}{\mathrm{F}}_\ell \\ \underset{\sim}{\tau}_\ell \end{bmatrix}, \tag{19}$$

and

$$\underset{\sim}{\Gamma}_\ell = \dot{\Phi}_\ell \underset{\sim}{\omega}_\ell + \Phi_\ell \mathrm{C}_\ell^{\mathrm{R}} \underset{\sim}{\xi}_\ell - I_v \underset{\sim}{g} , \qquad \underset{\sim}{\xi}_\ell = \widetilde{\omega}_\ell \underset{\sim}{\Omega}_\ell + \dot{\mathrm{D}}_\ell \dot{\underset{\sim}{\alpha}}_\ell . \tag{20}$$

Eqns.(10) and (13) can be used to express the EOM entirely in terms of true coordinates.

3.1 Different types of coordinate transformations

A large degree of flexibility is possible when using the transformation matrices, C_ℓ^R and D_ℓ. In this section, a few practical examples will be reviewed. One key observation is to note that the results of Section 2 do not change based upon the choice of the transformation matrices, and the explicit form of eqn.(18) does not change either. One requirement that must be made in formulating the AQEOM is that a sufficient number of body-fixed reference frames be used, such that any desired change in the system constraints does not require the introduction of a new reference frame. This requirement can be easily met by inspection of the system to be modeled. To illustrate this, consider the scenario in Figure 1.

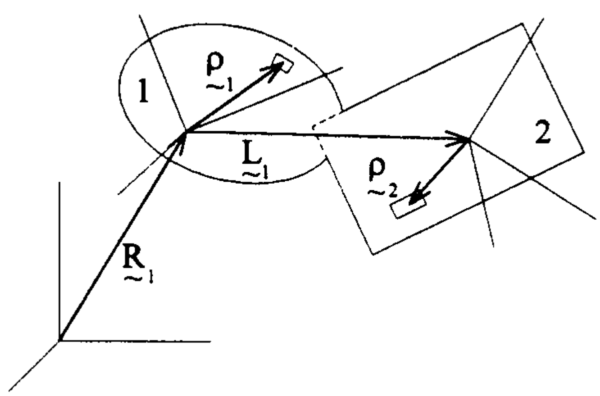

Figure 1: Flexibility of using multiple reference frames in a single rigid body

Shown here is one rigid body containing two separate coordinate systems, each being fixed with respect to the body. The dashed line indicates a boundary separating the body into two regions, the location of which is based on geometric considerations. The system is free to translate and rotate in three dimensional space, and frame 1 is chosen as the base frame. It is possible to represent the velocity, $\dot{\underset{\sim}{p}}_1$, of a differential mass in the first region of the body as

$$\dot{\underset{\sim}{p}}_1 = \dot{\underset{\sim}{R}}_1 + C_{N1}\tilde{\omega}_1 \underset{\sim}{\rho}_1, \tag{21}$$

where $\underset{\sim}{\rho}_i$ is the position of a differential mass in the i-region, expressed with respect to the i-frame. Also, the velocity of a differential mass in the second region of the body would be

$$\dot{\underset{\sim}{p}}_2 = \dot{\underset{\sim}{R}}_1 + C_{N1}\tilde{\omega}_1 \left[\underset{\sim}{L}_1 + C_{12} \underset{\sim}{\rho}_2 \right]. \tag{22}$$

By making the following substitution and using the subsequent identity,

$$C_{N1} = C_{N2}C_{21}, \qquad C\tilde{a}C^T = \widetilde{\left[C\underset{\sim}{a}\right]}, \tag{23}$$

$\underset{\sim}{\omega}_1$ can be expressed in the 2-frame as

$$C_{21}\underset{\sim}{\omega}_1 = \underset{\sim}{\omega}_2, \tag{24}$$

and by introducing eqn. (24) into eqn. (22), $\dot{\underset{\sim}{p}}_2$ can be expressed as

$$\dot{\underset{\sim}{p}}_2 = \dot{\underset{\sim}{R}}_1 + C_{N1}\tilde{\omega}_1 \underset{\sim}{L}_1 + C_{N2}\tilde{\omega}_2 \underset{\sim}{\rho}_2 . \tag{25}$$

Note that if the two regions where actually two separate bodies connected with a revolute joint, Eqns. (21) and (25) would be identical, and the results of Section 2 apply. Therefore, by including the second reference frame, we have allowed for the possibility of introducing (or removing) a revolute joint into the system by merely changing the D matrix as shown below.

3.1.1 Adding and removing DOFs in a limb with the D_ℓ matrix

What could be called "joint locking" is accomplished by making a simple change to the D_ℓ matrix for the limb ℓ. The collocated, independent actuation case is as D_ℓ appears in eqn.(14). If the joint between body (j-1) and body j, which has k_j DOFs, is to be locked, D_ℓ takes the following form:

$$D_\ell = \begin{bmatrix} D_1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & D_{j-1} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & D_{j+1} & \cdots & 0 \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & D_n \end{bmatrix}_{\ell,(3n\,x\,k')} , \tag{26}$$

where $k' = k - k_j$. Also, to reflect this change, the appropriate DOF in $\underset{\sim}{\alpha}_\ell$ is removed, reducing its size from (kx1) to $(k'\,x1)$.

It should be noted that if actual joint locking, through a clutching mechanism or brake, is to be simulated, then it may be desirable to also model the transient dynamics caused by this sudden change. A simple momentum balance applied at the joint could be used to generate simulated transient torques.

3.1.2 Coupling of joints in a limb using the D_ℓ matrix

If two (or more) joints, which have the same number of DOFs, are driven such that they share the same actuators, another slight modification to D_ℓ can capture this constraint. As an example, suppose that the otherwise independent motion of body j and body (j+1) are coupled kinematically, such that a single DOF moves both bodies. The resulting form of D_ℓ would be

$$\mathbf{D}_\ell = \begin{bmatrix} \mathbf{D}_1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & \mathbf{D}_j & \cdots & 0 \\ 0 & \cdots & \mathbf{D}_{j+1} & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & \mathbf{D}_n \end{bmatrix}_\ell, \tag{27}$$

where both $\mathbf{D}_j$ and $\mathbf{D}_{j+1}$ have dimension (3x1), and are chosen appropriately to represent the type of constrained joints being actuated. For instance, a pair of planar adjacent bodies may be connected such that their rotations are equal and opposite about the local z-axis. In such a case, we could use

$$\mathbf{D}_j^T = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}, \qquad \mathbf{D}_{j+1}^T = \begin{bmatrix} 0 & 0 & -1 \end{bmatrix}. \tag{28}$$

Of course, many other types of constraints of this form could be introduced. And, the constrained bodies need not be adjacent, as long the appropriate form for $\mathbf{D}_\ell$ is used.

3.1.3 Changing DOF definition using $\mathbf{C}_\ell^R$

Consider a case in which an internal mechanism causes a joint to be driven by an actuator which is inboard (towards the base frame) of the joint. This is equivalent to redefining a given joint angle (or angles) such that it is measured from an inboard body other than the most adjacent one. Cable driven joints would be an example of such a situation, wherein a motor (or motors) on body (j-2) drives the joint between bodies (j-1) and j. In such a case, we could redefine the joint angle between bodies (j-1) and j to represent the true DOF, which is at the driving motor on body (j-2). This is accomplished by setting locations in $\mathbf{C}_\ell^R$ which would normally have rotational transformations *from* the (j-1)-frame ($\mathbf{C}_{x(j-1)}$, where $j \le x \le n$)), equal to 0,

$$\mathbf{C}_\ell^R = \begin{bmatrix} I & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \mathbf{C}_{21} & I & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \mathbf{C}_{31} & \mathbf{C}_{32} & \ddots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ \vdots & \vdots & \ddots & I & 0 & 0 & \cdots & 0 & 0 \\ \mathbf{C}_{(j-1)1} & \mathbf{C}_{(j-1)2} & \cdots & \mathbf{C}_{(j-1)(j-2)} & I & 0 & \cdots & 0 & 0 \\ \mathbf{C}_{j1} & \mathbf{C}_{j2} & \cdots & \mathbf{C}_{j(j-2)} & 0 & I & \cdots & 0 & 0 \\ \mathbf{C}_{(j+1)1} & \mathbf{C}_{(j+1)2} & \cdots & \mathbf{C}_{(j+1)(j-2)} & 0 & \mathbf{C}_{(j+1)j} & \ddots & \vdots & \vdots \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & I & 0 \\ \mathbf{C}_{n1} & \mathbf{C}_{n2} & \cdots & \mathbf{C}_{n(j-2)} & 0 & \mathbf{C}_{nj} & \cdots & \mathbf{C}_{n(n-1)} & I \end{bmatrix}_\ell. \tag{29}$$

It is also possible to zero out more columns in $\mathbf{C}_\ell^R$, which would, in turn, further redefine the DOFs in the system.

3.1.4 Summary

It is apparent that there is great flexibility in defining the specific coordinate transformations that derive absolute angular velocities from generalized coordinates. Although this process is simply a redefining of the coordinates used to describe a system, it is important that the generalized coordinates be *independent* variables. Many times, this choice is aided by the kinematics of the system, as will be demonstrated in Section 5. Also, since D_ℓ is a transformation acting on velocities, transmission kinematics can be incorporated by including ratio values at the appropriate locations of this matrix. Likewise, a change in driven direction, such as is due to clutching, can be incorporated by sign changes in various places in D_ℓ based on which joints are being clutched.

4 Working with multiple interrelated limbs with constraints

We can now expand upon the above inferences to model complex systems. It should be noted that in general practice, the following algorithms can be largely condensed. We undertake the following discussion for mostly conceptual purposes.

As the term 'limb' suggests, the above discussion was an effort to describe the dynamics of a piece of a larger system. We will now investigate situations in which several limbs are connected to form a multibody system with various constraints. Consider such a system that contains L limbs, which are presently unconnected to each other. As described earlier, each limb ℓ is a multibody chain, where the bodies are connected with revolute joints. The 1 body of each limb contains a base reference frame that is free to translate and rotate in three dimensional space. Each limb is numbered with a Roman numeral, ℓ = I,II,...L.

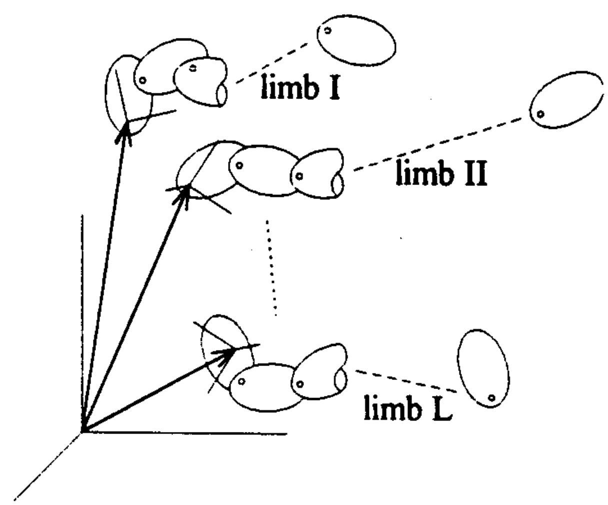

Figure 2: Global system modeled using dynamic limbs with constraints

So far, we have worked with a single limb, and introduced the following coordinate transformation:

$$\underset{\sim}{\overline{\omega}}_\ell = \overline{C}_\ell^R \overline{D}_\ell \underset{\sim}{\dot{\overline{\alpha}}}_\ell, \tag{30}$$

where

$$\bar{\underset{\sim}{\omega}}{}_{\ell}^{T} = \left[\dot{\underset{\sim}{R}}{}_{1\ell}^{T} \;\vdots\; \underset{\sim}{\omega}{}_{\ell}^{T}\right], \tag{31}$$

$$\bar{\underset{\sim}{\alpha}}{}_{\ell}^{T} = \left[\underset{\sim}{R}{}_{1\ell}^{T} \;\vdots\; \underset{\sim}{\alpha}{}_{\ell}^{T}\right], \quad \underset{\sim}{\alpha}{}_{\ell}^{T} = \left[\underset{\sim}{\alpha}{}_{1\ell}^{T} \;\vdots\; \underset{\sim}{\alpha}{}_{\ell}^{\prime T}\right], \quad \underset{\sim}{\alpha}{}_{\ell}^{\prime T} = \left[\underset{\sim}{\alpha}{}_{2\ell}^{T} \;\vdots\; \cdots \;\vdots\; \underset{\sim}{\alpha}{}_{n\ell}^{T}\right]. \tag{32}$$

We can concatenate these separate limbs into one large global system. Each limb has EOM as defined above, such that we can introduce the following global coordinate transformation:

$$\underset{\sim}{\omega}_{G} = C_{0}^{R} D_{0} \dot{\underset{\sim}{\alpha}}_{G0}, \tag{33}$$

where

$$\underset{\sim}{\omega}{}_{G}^{T} = \left[\bar{\underset{\sim}{\omega}}{}_{I}^{T} \;\vdots\; \bar{\underset{\sim}{\omega}}{}_{II}^{T} \;\vdots\; \cdots \;\vdots\; \bar{\underset{\sim}{\omega}}{}_{L}^{T}\right], \qquad \underset{\sim}{\alpha}{}_{G0}^{T} = \left[\bar{\underset{\sim}{\alpha}}{}_{I}^{T} \;\vdots\; \bar{\underset{\sim}{\alpha}}{}_{II}^{T} \;\vdots\; \cdots \;\vdots\; \bar{\underset{\sim}{\alpha}}{}_{L}^{T}\right], \tag{34}$$

$$C_{0}^{R} = \begin{bmatrix} \bar{C}_{I}^{R} & 0 & \cdots & 0 \\ 0 & \bar{C}_{II}^{R} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \bar{C}_{L}^{R} \end{bmatrix}, \qquad D_{0} = \begin{bmatrix} \bar{D}_{I} & 0 & \cdots & 0 \\ 0 & \bar{D}_{II} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \bar{D}_{L} \end{bmatrix}. \tag{35}$$

This transformation simply gives us all of the quasicoordinates for the entire system ('G' for global), in the vector $\underset{\sim}{\omega}_{G}$, from all of the true coordinates of the many limbs, $\underset{\sim}{\alpha}_{G0}$.

We can now introduce our first global constraint, that being that all of the limbs actually share a single, common base frame. This can also be described by indicating that all of the first, or 1 bodies of the L limbs are actually the same body in the system to be modeled.

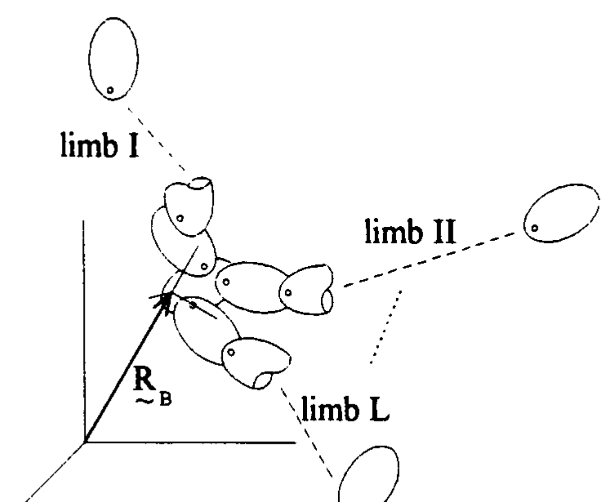

Figure 3: Example of typical constraint, a common base frame

We can therefore introduce the following additional coordinate transformation on the global system:

$$\underset{\sim}{\omega}_G = C_0^R D_0 D_1 \dot{\underset{\sim}{\alpha}}_{G1}, \tag{36}$$

where

$$\underset{\sim}{\alpha}_{G1}^T = \left[\underset{\sim}{R}_B^T \;\vdots\; \underset{\sim}{\alpha}_B^T \;\vdots\; \underset{\sim}{\alpha}_I'^T \;\vdots\; \underset{\sim}{\alpha}_{II}'^T \;\vdots\; \cdots \;\vdots\; \underset{\sim}{\alpha}_L'^T \right], \tag{37}$$

and 'B' indicates the global base frame. An example form of D_1 is shown in Section 5. Matrices like D_1 in this application are analogous in many ways to connectivity matrices used in the finite element method to construct solids from individual elements.

Because we have now essentially connected the various base frames of the separate limbs, we have produced a new rigid body, which contains the global base frame for the entire system. This new body is a summation of the separate limb base frames, and will therefore have the following mass:

$$m_B = \sum_{\ell=I}^{L} m_{1\ell}. \tag{38}$$

But, in general, we are approaching the modeling problem with knowledge of m_B and not $m_{1\ell}$. In order to deal with this, consider the following. The EOM are linear in mass, such that each term in the EOM contains the mass of one body only. Secondly, the EOM for body i in a chain of n bodies contains masses from outboard bodies only, that is bodies i, i+1, i+2,..., n. Therefore, we can arbitrarily choose to divide the mass of the global base frame body evenly among the separate limb base frame bodies,

$$m_{1\ell} = \frac{1}{L} m_B. \tag{39}$$

This is called 'mass partitioning', and a similar scheme should be used when dealing with $\underset{\sim}{F}_\ell$ and $\underset{\sim}{\tau}_\ell$. Using mass partitioning leads to algorithmic simplicity, and, when using the dynamic limb approach to model a given system, it can be summarized in most cases by the following general rule: the mass of a given body is partitioned by the number of limbs in which it is a common body. This is demonstrated well in Section 5.

It is sometimes convenient to introduce further transformations which reduce the number of generalized coordinates to a final set, $\underset{\sim}{\alpha}_{G3}$:

$$\underset{\sim}{\omega}_G = C_0^R D_0 D_1 D_2 D_3 \dot{\underset{\sim}{\alpha}}_{G3}. \tag{40}$$

Here, D_2 and D_3 are additional transformations which could possibly connect a subsystem of bodies together, or even introduce constraints from limb to limb. The following section will describe an example of these types of transformations.

5 Example: Modeling a Constrained Walking Robot

As an example of the above formulation, consider the following unique walking robot, which we will call the K^2T Walker (all diagrams courtesy K^2T, Inc.):

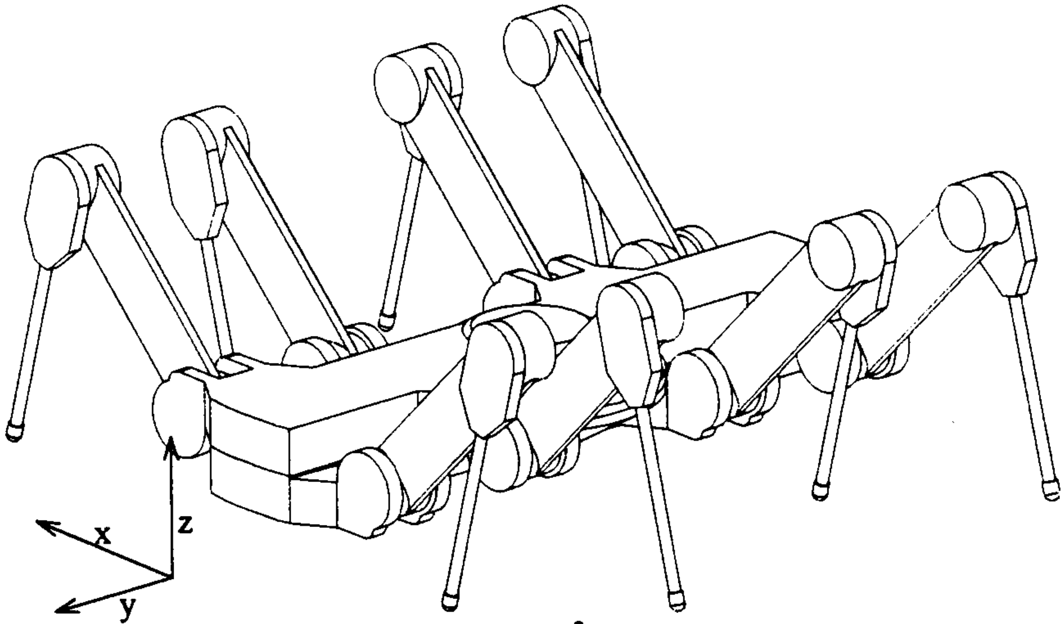

Figure 4: The K^2T Walker

5.1 Description of the K^2T Walker

The K^2T Walker is an eight legged walking robot which uses two bodies (or frames) which have been placed one on top of the other. Each frame contains four 2 DOF legs in a staggered configuration as shown. The two frames of the body are able to rotate relative to each other as shown in Figure 5. Also, each leg of the robot consists of two segments, a proximal segment called the "femur" and a distal segment referred to as the "tibia". Each leg uses two revolute joints, the axes of which are parallel:

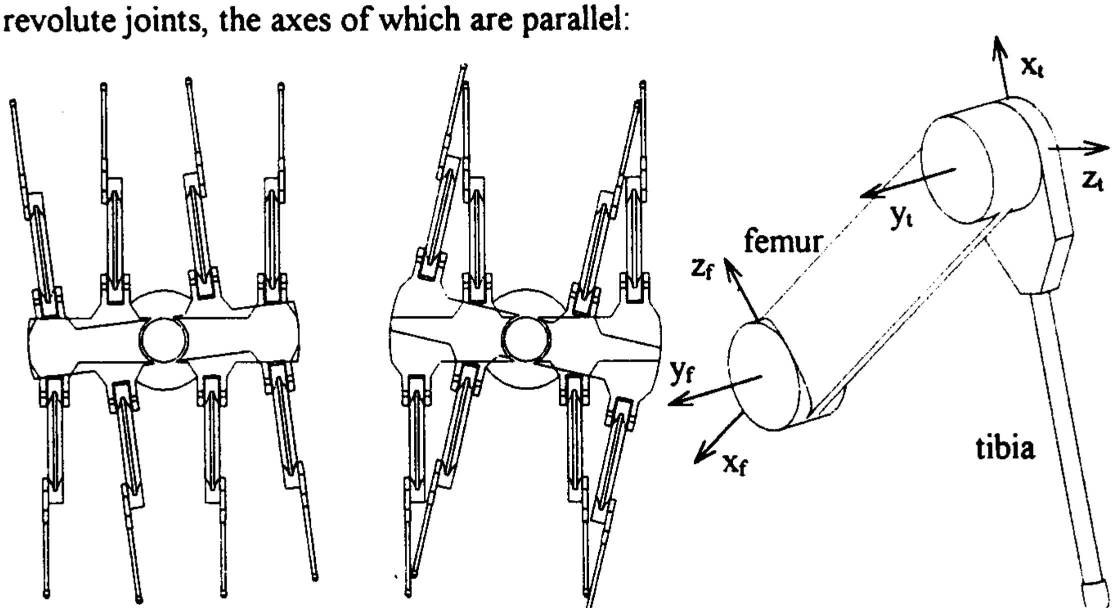

Figure 5: Top view showing body rotation. A rear leg of the K^2T Walker

As shown in Figure 4, an inertial coordinate system is chosen, with the x-axis being in the initial direction of travel, the y-axis being initially transverse to the robot, and the z-axis defining the vertical. Also, the two frames or bodies of the robot have their own body-fixed reference frames located at their respective mass centers. In the particular position shown in Figure 4, these body frames are parallel to the inertial N-frame.

5.2 Actuation of the K^2T Walker

Although the K^2T Walker possesses 17 separate joints, only 5 independent motors will be used to actuate the robot. All of these motors are located on the body of the robot. The first will control the rotation between the top and bottom bodies, or frames, of the robot. A second motor will drive the four femurs of one frame, and a third motor will stroke the four tibias of the same frame. Likewise, the remaining two motors will drive the femurs and tibias of the opposing frame in a similar manner.

Given the above, and according to the design of the robot, we can also indicate (for the sake of this example) that as the second or fourth motors rotates negatively, they will lift both the front and back femurs of their respective frames. Conversely, as these motors rotate positively, the femurs will be lowered. The third and fifth motors will move the tibias of their respective frames in a swinging motion, providing horizontal propulsion for the robot.

For simplicity, we will assume for this example that all joints are driven directly from the motors, and that transmissions are not being used. As discussed above, transmission ratios can be incorporated into the formulation at specific locations in the various D matrices. Clutching, though, is the primary means by which the robot performs nontrivial maneuvers, such as walking on rough terrain. This, therefore, is to be included in the model.

5.3 Modeling the K^2T Walker using dynamic limbs

Using the concept of a dynamic limb, we find that the K^2T Walker is composed of 8 limbs. We will use 5 generalized coordinates, corresponding directly to the separate virtual motor angles, to represent the system.

We can use the top frame of the robot as a base reference frame for the entire system. Therefore, four of these dynamic limbs are composed of (top body - bottom body - femur - tibia) chains, and the remaining four limbs are composed of (top body - femur -tibia) chains. For the purpose of mass partitioning, we see that the top body of the robot is common to all 8 limbs, while the bottom body is common to 4 limbs. Thus, we can partition their masses accordingly:

$$m_{1\ell} = \frac{1}{8} m_{\text{top body}}, \ \ell = 1,2,..8, \qquad m_{2\ell} = \frac{1}{4} m_{\text{bottom body}}, \ \ell = 1,2,3,4. \qquad (41)$$

Given that the properties of the robot, such as masses, inertias, and lengths, are known, we are left to describe the transformation matrices which will be used. Doing so will completely describe the EOM for this robot.

5.3.1 Basic EOM for normal locomotion

Given the constraints of the system, we can use the following D_ℓ matrices for the limbs of the system

$$D_{\substack{top\\limb}} = \left[\begin{array}{c:c:c} D_1' & 0 & 0 \\ \hdashline 0 & D_2' & 0 \\ \hdashline 0 & 0 & D_3' \end{array}\right]_{(9\times5)}, \quad D_{\substack{bottom\\limb}} = \left[\begin{array}{c:c:c:c} D_1' & 0 & 0 & 0 \\ \hdashline 0 & D_2'' & 0 & 0 \\ \hdashline 0 & 0 & D_3'' & 0 \\ \hdashline 0 & 0 & 0 & D_4'' \end{array}\right]_{(12\times6)}, \tag{42}$$

$$D_2' = D_3' = D_3'' = D_4'' = \begin{bmatrix}0 & 1 & 0\end{bmatrix}^T, \qquad D_2'' = \begin{bmatrix}0 & 0 & 1\end{bmatrix}^T, \tag{43}$$

and D_1' is a (3x3) matrix determined by an ordered set of Euler rotations representing the orientation of the base frame.

Recalling that the motors for both the femurs and tibias are located on the body, we can use the scheme described by eqn.(29), such that the tibia angle is measured relative to the body. As the design of the robot specifies, the tibias should passively remain fixed relative to the body when the corresponding motors are not activated, regardless of femur motion.

Since the 8 limbs share a common base frame, we can present the D_1 matrix, as discussed in Section 4, for conceptual purposes:

$$D_1^T = \left[D_A^T \;\vdots\; D_B^T\right], \tag{44}$$

$$D_A^T = \left[\begin{array}{c:c:c:c:c:c:c:c} I_{6x6} & 0_{6x3} & I_{6x6} & 0_{9x3} & I_{6x6} & 0_{12x3} & I_{6x6} & 0_{15x3} \\ 0_{20x6} & I_{3x3} & 0_{20x6} & I_{3x3} & 0_{20x6} & I_{3x3} & 0_{20x6} & I_{3x3} \\ & 0_{17x3} & & 0_{14x3} & & 0_{11x3} & & 0_{8x3} \end{array}\right], \tag{45}$$

$$D_B^T = \left[\begin{array}{c:c:c:c:c:c:c:c} I_{6x6} & 0_{18x2} & I_{6x6} & 0_{20x2} & I_{6x6} & 0_{22x2} & I_{6x6} & 0_{24x2} \\ 0_{20x6} & I_{2x2} & 0_{20x6} & I_{2x2} & 0_{20x6} & I_{2x2} & 0_{20x6} & I_{2x2} \\ & 0_{6x2} & & 0_{4x2} & & 0_{2x2} & & \end{array}\right]. \tag{46}$$

Likewise, we can use another transformation, D_2, to account for the fact that the bottom 4 limbs contain the same second body. We present it here, again for conceptual purposes:

$$D_2^T = \left[\begin{array}{c:c:c:c:c:c:c:c:c:c} I_{6x6} & 0_{6x1} & 0_{7x2} & 0_{6x1} & 0_{9x2} & 0_{6x1} & 0_{11x2} & 0_{6x1} & 0_{13x2} & 0_{15x8} \\ 0_{17x6} & 1 & I_{2x2} & 1 & I_{2x2} & 1 & I_{2x2} & 1 & I_{2x2} & I_{8x8} \\ & 0_{16x1} & 0_{14x2} & 0_{16x1} & 0_{12x2} & 0_{16x1} & 0_{10x2} & 0_{16x1} & 0_{8x2} & \end{array}\right]. \tag{47}$$

The system has now been reduced to 23 DOFs, consisting of 17 joint angles and the 6 rigid body DOFs for the base frame. We can now introduce a final transformation, D_3, which couples the femurs and tibias of the various legs as described above. This is shown in eqns.(48) and (49). Clutch modeling is achieved by making the necessary sign changes in the individual leg D matrices. Also, the same result could be reached by making sign changes in the other global D matrices. For instance, to change the driven direction of the femurs, sign changes could be made in the D_F matrix below. A simple momentum model is used to generate the transient forces caused by the clutching action.

$$D_3^T = \begin{bmatrix} I_{6x6} & 0_{6x1} & 0_{7x8} & 0_{9x8} \\ 0_{5x6} & 1 & D_F & D_F \\ & 0_{4x1} & D_T & D_T \\ & & 0_{2x8} & \end{bmatrix}, \tag{48}$$

$$D_F = [1 \quad 0 \quad 1 \quad 0 \quad -1 \quad 0 \quad -1 \quad 0], \quad D_T = [0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1]. \tag{49}$$

5.3.2 Simulation

In practice, the global transformations, D_0, D_1, D_2, D_3, can be multiplied together off-line to produce single transformations from the final generalized coordinates to the absolute angular velocities for each limb. Thus, a single computational loop across the limbs can apply eqn.(18) using the appropriate transformations, and the final global EOM can be constructed easily. The C++ programming language was used to code a simulation of this robot. A force model was used for simulating the ground, which is discussed in Nelson[12], adding flexibility to the simulation with friction models for the ground.

6 Conclusion

This paper presents a solution to the general multibody dynamics problem rather than simply a formulation. The equations of motion representing this solution are explicit and their form is intuitive. Kinematic constraints can be readily introduced such that the system order is reduced. In addition, the solution is most useful for design because the model can be quickly changed so that results can be readily compared with competing designs. The tradeoff is that the solution is not "order-n". However, our experience with systems of as many as 36 degrees of freedom is that the computational speed on personal computers is adequate for mechanical design and testing control strategies. A model has been developed for the K^2T Walker and has been used with great success for its design.

Acknowledgments

The authors would like to thank K^2T, Inc., and the Office of Naval Research for supporting this work. Mr. Nelson is supported by grant NGT-51194 through the NASA Marshall Space Flight Center.

References

1. Hooker, W., and Margulies, G., "The Dynamical Attitude Equations for a n-Body Satellite," Vol. 12, pp. 123-128, *Journal of Astronautical Sciences*, 1965.

2. Roberson, R.E., and Wittenburg, J., 1967, "A Dynamical Formalism for an Arbitrary Number of Interconnected Rigid Bodies with Reference to the Problem of Satellite Attitude Control," pp. 45D.1-46D.8, *Proceedings of the Third International Congress on Automatic Control*, Butterworth, London, England, 1967.

3. Frisch, H.P., "A Vector-Dyadic Development of the Equations of Motion for N-Coupled Rigid Bodies and Point Masses," NASA Technical Note TN D-7767, 1974.

4. Bodley, C.A., Devers, A., Park, A. and Frisch, H.P., "A Digital Computer Program for Dynamic Interaction Simulations of Controls and Structures (DISCOS)," NASA Technical Paper 1219, 1978.

5. Roberson, R.E., and Schwertassek, R.F., *Dynamics of Multibody Systems*, Springer-Verlag, Berlin, 1987.

6. Amirouche, F.M.L., *Computational Methods in Multibody Dynamics*, Prentice Hall, Englewood Cliffs, NJ., 1992.

7. Brandl, H., Johanni, R., and Otter, M., "A Very Efficient Algorithm for the Simulation of Robots and Similar Multibody Systems Without Inversion of the Mass Matrix," *Proceedings of the IFAC/IFIP/IMACS International Symposium of the Theory of Robots*, Vienna, Austria, 1986.

8. Lilly, K.W., and Orin, D.E., "Efficient Dynamic Simulation of Multiple Chain Robotic Mechanism," Vol.116, pp.223-231, *Journal of Dynamic Systems, Measurement, and Control*, June, 1994.

9. Meirovitch, L., *Methods of Analytical Dynamics*, McGraw-Hill Publishing Company, New York, NY., 1970.

10. Quinn, R.D., "Equations of Motion for Structures in Terms of Quasicoordinates," Vol.57, No.3, pp.745-749, *Journal of Applied Mechanics*, 1990.

11. Quinn, R.D., and Chang, R.Y.K., "Dynamic Modeling of Multi-Jointed Mechanical Systems for Space Applications," *Proceedings of the 7th VPI&SU/AIAA Symposium on Dynamics and Control of Large Structures*, May, 1989.

12. Nelson, G.M., "Modeling and Simulation of an Insect-like Hexapod," M.S. Thesis, Case Western Reserve University, 1995.

13. Nelson, G.M., and Quinn, R.D., "A Lagrangian Quasicoordinate Formulation for Dynamic Simulations of Multibody Systems," pp. 823-830, Vol. 57-2, *Proceedings of the ASME Dynamic Systems and Control Division*, San Francisco, California, 1995.