ROBOT DYNAMICS

When a robotic system, such as a robotic arm (or a manipulator), moves in a slow motion, then no appreciable dynamic load will be observed between its links. Hence, under such a condition, only static equilibrium conditions will be sufficient to model and control the robot. For instance, consider a human arm representing a robotic manipulator. In order to hold the arm in equilibrium in an outward-extended position, a certain amount of effort must be spent at each joint so the weight of the arm is balanced. In a mechanical arm, this static equilibrium situation is replicated by applying a certain amount of torque at each joint by an actuator.

If a robotic arm undergoes rapid movements, then interactions between links due to mass content of the links and payload may not be neglected since such movements cause additional loads on the system known as inertial effects. This may be experienced in a human arm when the arm is moved very quickly from one position to another. One can feel the extra load during this rapid movement and the difficulty to instantly stop it. This additional load, experienced in quick movements, which is nonexistent when the arm moves slowly, is the inertial effect one needs to consider in order to model the robot accurately.

An accurate model ensures accurate calculation and simulation of robot motions on the computer before the actual motion takes place. Also, a good model helps the engineer in the design stage by providing an opportunity to test performance of the robot before the design is finalized. Robot models are also useful in developing control algorithms. After the control strategy is planned, simulation on the computer again provides valuable insight on the effectiveness of a certain control strategy even before it is tested on the actual robot. Hence, one can conclude that a good robot model is an extremely useful tool for the engineer who is responsible for designing a robot or developing software to control it.

Before reviewing robot dynamics below, the notation used for geometric description in robotics, and kinematic relations are briefly introduced. The kinematic formulation, originally developed for three-dimensional mechanisms, has become standardized to the form presented by Denavit and Hartenberg in Ref. 1. Dynamic modeling techniques include the Newton–Euler method, the Lagrange method, and Lagrange's form of the Generalized Principle of d'Alembert. In general, the Newton–Euler method provides better computational efficiency, but the Lagrangian dynamics provide more physical insight in the development, and in implementation the method has become comparable to the Newton–Euler method. Some of the work in this area may be found in Refs. 2–6. The method presented here is the Lagrangian form of Generalized Principle of d'Alembert, based on the notation presented in Ref. 7, and extended later to flexible systems in Ref. 8.

ROBOT KINEMATICS

Kinematics is a branch of dynamics that studies motion without any consideration of mass or force. For our purposes, this means a study of position, velocity, and acceleration of the robot. More precisely, in robotics, this study represents a mapping between the so-called joint space and the end-effector space. This is justified, because in practical applications, the robot's end-effector motion is almost always specified, but the robot is powered through its joints. Hence, a mapping, or a relationship, is needed between the joint and end-effector spaces. This is necessary at the position, velocity, and acceleration levels.

A serial manipulator or a robotic arm is treated as an open-chain linkage, most often composed of one degree of freedom (DOF) revolute and prismatic joints. A revolute joint allows a rotational motion between the two links that it connects. This is similar to the elbow joint in a human arm. A prismatic joint, on the other hand, allows only a sliding motion between the links that it connects.

There are two main issues to be considered in kinematics: forward and inverse kinematics. In the forward kinematics case, given the robot dimensions and the joint displacements, velocities, and accelerations, we obtain the end-effector translational and rotational displacement, velocity, and acceleration. In the inverse kinematics case, we obtain the joint displacements, velocities, and accelerations when given the end-effector kinematic conditions.

Geometric Parameters

A serial manipulator is generally considered as a set of links in an open kinematic chain connected through one-DOF joints. The standard kinematic notation for a general manipulator uses the Denavit–Hartenberg (D–H) parameters, and a set of reference frames as shown in Fig. 1. A fixed Cartesian coordinate frame is placed at the base joint with the z-axis pointing along the same direction as the joint 1 axis. Each joint has a local coordinate frame attached to it. The fixed absolute frame is normally chosen to coincide with the frame of the first joint's local frame, only separated by an offset. The orientation of each link is defined by the values of the unit direction cosine vectors, $S_j \in \mathscr{R}^3$ and $a_{jk} \in \mathscr{R}^3$, which respectively define the local z_j- and x_j-axes. The vector S_j defines the joint axis, either the axis of rotation for revolute joints, or the axis of displacement for prismatic joints. The vector a_{jk} defines the direction of the mutual perpendicular line for neighboring links j and k.

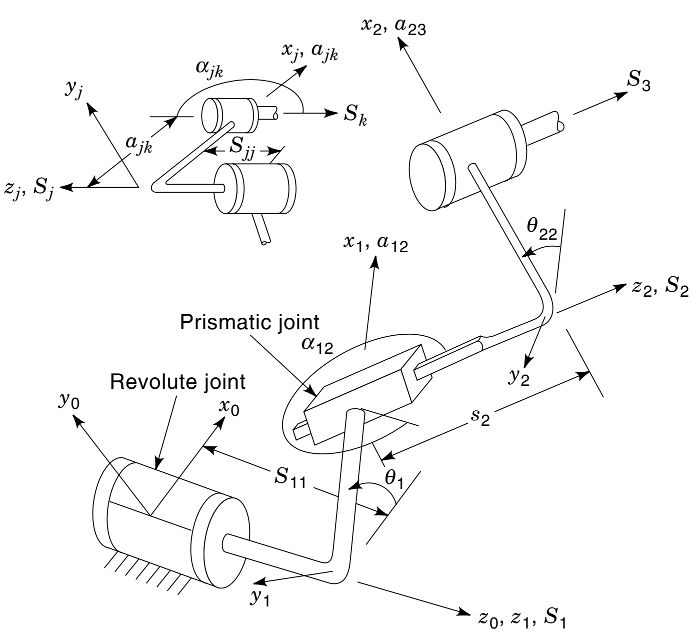

Figure 1. General kinematic representation of robotic manipulators with revolute and prismatic joints.

Table 1. Summary of Kinematic Parameters

Parameter	Definition
Joint Axis, S_j	Direction cosine vector of joint j, coincident with z_j
Link Axis, a_{jk}	Direction cosine vector of link jk, coincident with x_j
Link Length, a_{jk}	Distance from z_j to z_k measured along x_j
Joint Offset, S_{jj}	Fixed distance from x_i to x_j measured along z_j for revolute joint
Joint Offset, θ_{jj}	Fixed angle between x_i and x_j measured about z_j for prismatic joint
Link Twist Angle, α_{jk}	Fixed angle between z_j and z_k measured about x_j
Joint Angle, θ_j	Variable angle between x_i and x_j measured about z_j for revolute joint
Joint Displacement, s_j	Variable distance from x_i and x_j measured along z_j for prismatic joint

Each joint has an associated constant offset S_{jj} for revolute, or θ_{jj} for prismatic joint, with respect to the joint axis S_j. Each joint also has a joint variable s_j for prismatic, θ_j for revolute, or z_j for the general case, along/about the joint vector S_j. In the notation, double subscripts represent constants, and single subscripts represent joint variables.

Although the n links in an n-DOF robotic arm will be numbered from 1 towards n later on, we will now consider that links are numbered as 12, 23, . . ., jk, . . ., $n(n + 1)$ as was done earlier in developing the geometric parameters for robotic systems.

Each link jk thus defined possesses a pair of constant parameters: (1) a_{jk}, which is the magnitude of the perpendicular distance between the joint axes measured along the common perpendicular vector a_{jk}; and (2) α_{jk}, the twist angle between the joint axes measured in a right-hand sense about the vector a_{jk}. Table 1 summarizes the representation. A simple procedure to follow in assigning the link frames to the joints as suggested in Ref. (9) is given as follows:

1. Identify the joint axes 1 through n for a robot with n joints (here such a robot is also assumed to be n-DOF).
2. Consider two neighboring (adjacent) joints denoted by j and k, and identify their joint axes, S_j and S_k.
3. Identify the common perpendicular, a_{jk}, between them. At the point of intersection of S_j and a_{jk}, assign the link frame origin, O_j.
4. Assign the z_j-axis pointing along the jth joint axis, S_j.
5. Assign the x_j-axis pointing along the common perpendicular, a_{jk}.
6. Assign the y_j-axis in a right-hand rule sense such that $x_j \times y_j = z_j$.
7. Repeat steps 2 through 6 for all intermediate links.

For simplicity, assign the fixed frame 0 to match frame 1 when the first joint variable is zero. The last frame can be chosen arbitrarily but it should be selected to zero out some

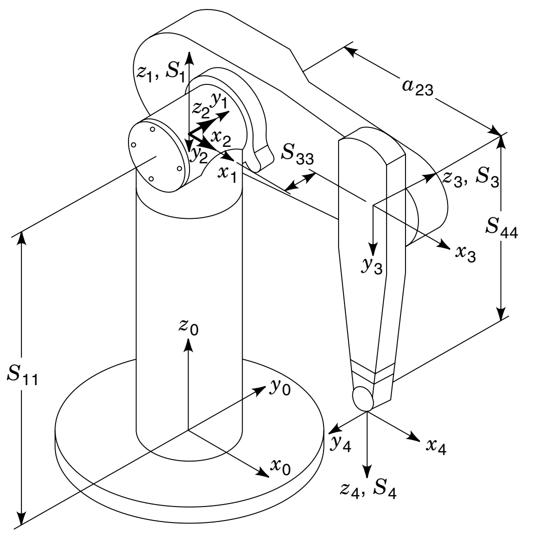

Figure 2. Kinematic parameters and frames on a 4-DOF industrial robot.

parameters for simplicity. This selection of reference frames is illustrated in Fig. 2 on a 4-DOF industrial robot which uses four revolute joints in its design.

Transformation Matrices

Rotational transformation matrices are needed to relate the local coordinate frame of joint k to the local coordinate frame j. Such transformations from the end-effector to the absolute coordinate frame are needed to formulate the kinematics problem, i.e., to determine the mapping between joint displacements and end-effector location. Transforming one frame to another involves four individual transformations; two rotations, and two translations. From the D-H parameters presented in the previous section, we see that adjacent frames are related by an x-axis rotation of "α," an x-axis translation of "a," a z-axis rotation of "θ," and a z-axis translation of "S." Thus, frame k may be transformed to frame j (frame k expressed in frame j) by

$$ {}^{j}_{k}T = R_x(\alpha_{jk})D_x(a_{jk})R_z(\theta_k)D_z(S_{kk}) \in \mathscr{R}^{4\times 4} $$

$$ {}^{j}_{k}T = \begin{bmatrix} \cos\theta_k & -\sin\theta_k & 0 & a_{jk} \\ \sin\theta_k\cos\alpha_{jk} & \cos\theta_k\cos\alpha_{jk} & -\sin\alpha_{jk} & -\sin\alpha_{jk}S_{kk} \\ \sin\theta_k\sin\alpha_{jk} & \cos\theta_k\sin\alpha_{jk} & \cos\alpha_{jk} & \cos\alpha_{jk}S_{kk} \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

where $R_x(\alpha_{jk})$ represents a rotation about the x-axis of α_{jk}, $D_x(\alpha_{jk})$ stands for a linear displacement along the x-axis of a_{jk}, and the same applies to the transformations with respect to the z-axis. With the assignment of the robot D–H parameters, the transformations can be computed. This transformation can be viewed as

$$ {}^{j}_{k}T = \left[\begin{array}{ccc:c} & [{}^{j}_{k}R] & & r^{j}_{ok} \\ \hdashline 0 & 0 & 0 & 1 \end{array} \right] $$

where $[{}^{j}_{k}R] \in \mathscr{R}^{3\times 3}$ represents the rotational part, i.e., describes frame k relative to frame j, and $r^{j}_{ok} \in \mathscr{R}^{3}$ is the expression for the origin of frame k, O_k, location in frame j. The individual transformations are compounded by multiplying them by the transformation matrix of an adjacent coordinate frame. Thus,

$$ {}^{i}_{k}T = {}^{i}_{j}T\,{}^{j}_{k}T $$

gives the transformation from frame i to frame k (frame k as seen from i), and

$$ {}^{0}_{n}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T \cdots {}^{n-1}_{n}T $$

gives the transformation between the last link and the absolute coordinate frame (frame n as seen from the fixed reference frame 0). Each individual transformation is a function of a single joint variable while the transformation, ${}^{0}_{n}T$, is a function of all joint variables that describes the end-effector location, position, and orientation through the joint variables.

Using these transformation matrices, various vector quantities may be represented in different reference frames. For instance, consider a joint vector S_j, which is expressed in the j'th reference frame as S^j_j. Also consider that the same vector is expressed in the fixed frame as S^0_j, or simply S_j. If the local representation S^j_j is known, then S_j is calculated from

$$ S_j = {}^{0}_{j}R S^{j}_{j} $$

On the other hand, if a position vector in frame j pointing to a point of interest is expressed as r^j_j, then the position vector to the same point from the fixed frame 0 is given by using the 4×4 transformation matrix as

$$ r_j = r^0_j = {}^{0}_{j}T r^{j}_{j} $$

The physical interpretation of these mappings defined by the rotation matrix ${}^{0}_{j}R \in \mathscr{R}^{3\times 3}$, and transformation matrix ${}^{0}_{j}T \in \mathscr{R}^{4\times 4}$ is illustrated in Fig. 3.

Robot Position, Velocity and Acceleration Expressions

In order to establish the kinematic relationship, let $r_{pq} \in \mathscr{R}^3$ represent the position vector from joint p to joint q; $r_{pcq} \in \mathscr{R}^3$ describe the position vector from joint p to the center of mass of link or payload q; and $S_p \in \mathscr{R}^3$ be the joint vector for joint p as described above. Then, the absolute position, translational velocity, and acceleration of the center of mass of link

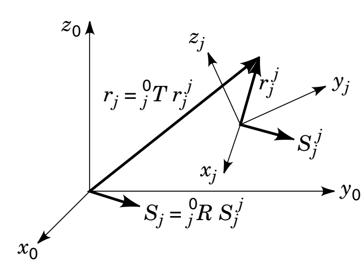

Figure 3. Effect of mappings via transformation and rotation matrices.

q in a global fixed frame are given, respectively, by

$$\begin{aligned} r_{ocq} &= r_{o1} + r_{12} + \cdots + r_{pq} + r_{qcq} \\ \dot{r}_{ocq} &= \dot{r}_{o1} + \dot{r}_{12} + \cdots + \dot{r}_{pq} + \dot{r}_{qcq} \\ \ddot{r}_{ocq} &= \ddot{r}_{o1} + \ddot{r}_{12} + \cdots + \ddot{r}_{pq} + \ddot{r}_{qcq} \end{aligned} \tag{1}$$

The translational velocity and acceleration expressions in the above equations are represented in terms of the position-dependent, translational g- and h-functions as

$$\begin{aligned} \dot{r}_{ocq} &= ({}_TG_q)\dot{z} \\ \ddot{r}_{ocq} &= ({}_TG_q)\ddot{z} + \dot{z}^T({}_TH_q)\dot{z} \end{aligned} \tag{2}$$

which are expanded as follows

$$\begin{aligned} \dot{r}_{ocq} &= \sum_{k=1}^{n} ({}_TG_q)_k \dot{z}_k \\ \ddot{r}_{ocq} &= \sum_{k=1}^{n} ({}_TG_q)_k \ddot{z}_k + \sum_{k=1}^{n}\sum_{j=1}^{n} ({}_TH_q)_{kj} \dot{z}_k \dot{z}_j \end{aligned} \tag{3}$$

where $\dot{z} \in \mathscr{R}^n$ and $\ddot{z} \in \mathscr{R}^n$ represent the joint velocities and accelerations, respectively. For instance, for an n-link robot with n revolute joints, $\dot{z}$ becomes

$$\begin{aligned} \dot{z} &= [\dot{z}_1 \quad \dot{z}_2 \dots \dot{z}_n]^T \\ &= [\dot{\theta}_1 \quad \dot{\theta}_2 \dots \dot{\theta}_n]^T \end{aligned} \tag{4}$$

In Eq. (2), ${}_TG_q = {}_TG_q(z) \in \mathscr{R}^{3\times n}$ and ${}_TH_q = {}_TH_q(z) \in \mathscr{R}^{3\times n\times n}$ represent the position-dependent, translational g- and h-functions for link q, and in reference to Eq. (3), $({}_TG_q)_k$ and $({}_TH_q)_{kj}$ are defined by

$$({}_TG_q)_k = \frac{\partial r_{ocq}}{\partial z_k} \in \mathscr{R}^3 \quad \text{and} \quad ({}_TH_q)_{kj} = \frac{\partial^2 r_{ocq}}{\partial z_k \partial z_j} \in \mathscr{R}^3$$

Similarly, the rotational velocity, ω_q, and acceleration, $\dot{\omega}_q$, of link q are given in terms of position-dependent, rotational g- and h-functions as

$$\begin{aligned} \omega_q &= ({}_RG_q)\dot{z} \\ \dot{\omega}_q &= ({}_RG_q)\ddot{z} + \dot{z}^T({}_RH_q)\dot{z} \end{aligned} \tag{5}$$

where ${}_RG_q \in \mathscr{R}^{3\times n}$ and ${}_RH_q \in \mathscr{R}^{3\times n\times n}$ have the same structure as their translational counterparts. Translational and rotational g-functions, $({}_TG_q)_k$, $({}_RG_q)_k$, and h-functions, $({}_TH_q)_{kj}$, $({}_RH_q)_{kj}$, are listed in Tables 2 and 3.

Table 2. Rotational g- and h-Functions ${}_RG_j$ and ${}_RH_j$

Joint k[a]	Joint i	$({}_RG_j)_i$	$({}_RH_j)_{ik}$	Conditions
R	R	S_i	$S_k \times S_i$	$k < i \le j$
R	R	S_i	0	$i \le j, k$
R	R	0	0	$j < i$, all k
P	R	S_i	0	$i \le j$, all k
P	R	0	0	$j < i$, all k
R	P	0	0	all i, j, k
P	P	0	0	all i, j, k

[a] R: Revolute, P: Prismatic joint.

Table 3. Translational g- and h-Functions ${}_TG_j$ and ${}_TH_j$

Joint k	Joint i	$({}_TG_j)_i$	$({}_TH_j)_{ik}$	Conditions
R	R	$S_i \times r_{icj}$	$S_k \times (S_i \times r_{icj})$	$k \le i \le j$
R	R	$S_i \times r_{icj}$	$S_i \times (S_k \times r_{kcj})$	$i \le k \le j$
R	R	$S_i \times r_{icj}$	0	$i \le j < k$
R	R	0	0	$j < i$, all k
P	R	$S_i \times r_{icj}$	0	$k < i \le j$
P	R	$S_i \times r_{icj}$	$S_i \times S_k$	$i < k \le j$
P	R	$S_i \times r_{icj}$	0	$i \le j < k$
P	R	0	0	$j < i$, all k
R	P	S_i	$S_k \times S_i$	$k < i \le j$
R	P	S_i	0	$i \le j, k$
R	P	0	0	$j < i$, all k
P	P	S_i	0	$i \le j$, all k
P	P	0	0	$j < i$, all k

ROBOT DYNAMIC EQUATIONS

In this section, the dynamic equations for an n-DOF serial robot described above are derived by employing the Lagrangian formulation. For this purpose, the total kinetic and potential energy expressions for the robot with n degrees of freedom are given as

$$KE = \sum_{j=1}^{n} (KE)_j \tag{6}$$

where the total kinetic energy of the robot is the total sum of kinetic energy of each of its links. If the kinetic energy of link j is denoted by $(KE)_j$, it is expressed as follows

$$\begin{aligned} (KE)_j = \frac{1}{2}\{&\dot{z}^T({}_TG_j)^T M_j ({}_TG_j)\dot{z} \\ &+ \dot{z}^T({}_RG_j)^T I_j ({}_RG_j)\dot{z}\} \end{aligned} \tag{7}$$

and

$$PE = (PE)_g \tag{8}$$

where M_j is the mass of link j, $I_j \in \mathscr{R}^{3\times 3}$ is the inertia tensor of body j about its center of mass, and $(PE)_g$ is the gravitational potential energy. The Lagrange equations, on the other hand, are defined by

$$\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{z}_i}\right) - \frac{\partial KE}{\partial z_i} + \frac{\partial PE}{\partial z_i} = \tau_i; \quad i = 1, 2, \dots, n \tag{9}$$

where τ_i is the effective actuator torque applied at joint i. If the kinetic energy expression in Eq. (6) is substituted into Eq. (9), the first two terms of the Lagrange equation which involve only the arm kinetic energy yield the following expression

$$\begin{aligned} \frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{z}_i}\right) - \frac{\partial KE}{\partial z_i} = \sum_{j=1}^{n} \{&({}_TG_j)_i^T M_j ({}_TG_j \ddot{z} + \dot{z}_T^T H_j \dot{z}) \\ &+ ({}_RG_j)_i^T [I_j ({}_RG_j \ddot{z} + \dot{z}_R^T H_j \dot{z}) \\ &+ ({}_RG_j \dot{z}) \times (I_{jR} G_j \dot{z})]\gamma_i\} \end{aligned} \tag{10}$$

where $\times$ indicates the vector cross product. The term γ_i in Eq. (10) is introduced to identify the joint type and is defined as

$$\gamma_i = \begin{cases} 1 & \text{if the } i\text{th joint is revolute} \\ 0 & \text{if the } i\text{th joint is prismatic} \end{cases} \tag{11}$$

If the ith joint is prismatic, then $\gamma_i = 0$ implies that only the first term on the right-hand side of Eq. (10) will be nonzero. Using Table 3 (the g-function $({}_TG_j)_i = S_i$ for prismatic joints when $i \le j$; 0 otherwise), and Eq. (2), which defines the absolute linear acceleration of the center of mass of a link ($\ddot{r}_{ocj} = ({}_TG_j)\ddot{z} + \dot{z}^T({}_TH_j)\dot{z}$), Eq. (10) reduces to

$$\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{z}_i}\right) - \frac{\partial KE}{\partial z_i} = \sum_{j=1}^{n} S_i \cdot (M_j \ddot{r}_{ocj})$$

which projects the inertial forces acting on link j, $(i \le j \le n + m)$, on prismatic joint i. A similar argument can be given if the ith joint is revolute. In this case, the translational g-function $({}_TG_j)_i$ in the first term of Eq. (10) is given by $S_i \times r_{icj}$ for $i \le j$; and it is 0 otherwise (Table 3). Noting that $a \times b \cdot c = a \cdot b \times c$ holds for three arbitrary vectors a, b, and c, and also using Eq. (2), the first term in Eq. (10) can be expressed as

$$\sum_{j=i}^{n} S_i \times r_{icj} \cdot (M_j \ddot{r}_{ocj}) = \sum_{j=i}^{n} S_i \cdot [r_{icj} \times (M_j \ddot{r}_{ocj})]$$

which projects the moment of translational inertia forces (acting on links i through n) on S_i. Finally, the second term on the right-hand side of Eq. (10) accounts for the rotational inertia effects. Since Table 2 defines $({}_RG_j)_i = S_i$ for $i \le j$, and Eq. (5) describes the absolute angular velocity and acceleration expressions in terms of g- and h-functions, this term is expressed as $\sum_{j=i}^{n} S_i \cdot [I_j\dot{\omega}_j + \omega_j \times I_j\omega_j]$ which is the projection of the rotational inertia of link j, $(i \le j \le n)$, on revolute joint i. Hence, for a revolute joint i, Eq. (10) reduces to

$$\frac{d}{dt}\left(\frac{\partial KE}{\partial \dot{z}_i}\right) - \frac{\partial KE}{\partial z_i} = \sum_{j=i}^{n} S_i \cdot [r_{icj} \times (M_j \ddot{r}_{ocj}) + I_j \dot{\omega}_j + \omega_j \times I_j \omega_j]$$

The contribution of potential energy due to gravitational effects is described by

$$\frac{\partial PE}{\partial z_i} = \sum_{j=1}^{n} ({}_TG_j)_i^T M_j g \tag{12}$$

In Eq. (12), g represents the gravitational acceleration vector which is expressed as $g = [g_x\ g_y\ g_z]^T$ in the global reference frame. The substitution of Eqs. (10) and (12) into Eq. (9) yields the robot dynamic equations

$$M(\theta)\ddot{\theta} + F(\theta, \dot{\theta}) = T(t) \tag{13}$$

in which $\ddot{\theta} = \dot{\omega} = \ddot{z} \in \mathscr{R}^n$ is the joint acceleration vector, $M(\theta) \in \mathscr{R}^{n\times n}$ is the generalized inertia matrix whose element M_{ij} is given by

$$M_{ij} = \sum_{q=1}^{n} \{({}_TG_q)_i^T M_q ({}_TG_q)_j + ({}_RG_q)_i^T I_q ({}_RG_q)_j\} \tag{14}$$

$F(\theta, \dot{\theta}) \in \mathscr{R}^n$ contains the centrifugal and Coriolis acceleration effects as well as the gravitational loads, and its element F_i is defined as

$$\begin{aligned} F_i = \sum_{q=1}^{n} \{ & ({}_TG_q)_i^T M_q (\dot{z}_T^T H_q \dot{z}) \\ & + ({}_RG_q)_i^T [I_q(\dot{z}_R^T H_q \dot{z}) + ({}_RG_q \dot{z}) \times (I_{qR} G_q \dot{z})]\gamma_i \\ & + ({}_TG_q)_i^T M_q g\} \end{aligned} \tag{15}$$

and $T(t)$ is the vector of input actuator torques/forces described as

$$T(t) = [\tau_1 \quad \tau_2 \ldots \tau_n]^T \in \mathscr{R}^n \tag{16}$$

for an n-DOF robot.

In terms of the input vector, $T(t)$, Eq. (13) represents a set of n algebraic equations. If joint positions, velocities, and accelerations are specified, then $T(t)$ may be calculated directly from Eq. (13). In terms of the joint position vector, θ, however, the same set of equations represents a set of n second-order, nonlinear, coupled, ordinary differential equations for which closed-form solutions for even the most simple robots do not exist.

It is important to note that for this model, the generalized inertia matrix $M(\theta)$ is positive definite over the entire workspace provided that the mass content of robot links or payload is nonzero. This result may be concluded from the robot's kinetic energy expression.

$$KE = \frac{1}{2}\dot{\theta}^T[M(\theta)]\dot{\theta} > 0 \quad \text{for all} \quad \dot{\theta} \neq 0$$

since it is always a positive quantity as long as at least one of the links has nonzero joint velocity. Positive definiteness of the M matrix implies its nonsingularity over the robot's entire workspace. Hence, we conclude by premultiplying Eq. (13) by the inverse of M that

$$\ddot{\theta} = [M(\theta)]^{-1}\{T(t) - F(\theta, \dot{\theta})\} \tag{17}$$

which is the form necessary for simulation purposes. This format is especially suitable to test various controllers numerically on the computer by evaluating system response. Utilizing known initial joint positions, velocities, and the actuator inputs described by the controller under evaluation, the above set of equations is numerically integrated to obtain the resulting robot motion.

BIBLIOGRAPHY

1. J. Denavit and R. S. Hartenberg, A kinematic notation for lower-pair mechanisms based on matrices, *J. Appl. Mech.,* 215–221, 1955.
2. J. Uicker, *On the dynamic analysis of spatial linkages using 4 × 4 matrices,* Ph.D. Dissertation, Northwestern Univ., Evanston, Illinois, 1965.
3. Y. Stepanenko and M. Vukobratovic, Dynamics of articulated open-chain active mechanisms, *Math-Biosciences,* **28**: 137–170, 1976.
4. M. Walker and D. Orin, Efficient dynamic computer simulation of robot mechanisms, *ASME J. Dynamic Sys., Meas., Control,* **104**: 1982.

5. J. M. Hollerbach, A recursive Lagrangian formulation of manipulator dynamics and comparative study of dynamics formulation complexity, in *Robot Motion,* M. Brady (ed.), Cambridge, MA: MIT Press, 1982.
6. W. Book, Recursive Lagrangian dynamics of flexible manipulator arms, *Int. J. Robotics Res.,* **3** (3): 87–101, 1984.
7. M. Thomas and D. Tesar, Dynamic modeling of serial manipulator arms, *ASME J. Dynamic Sys., Meas., Control,* **104**: 218–228, 1982.
8. S. Tosunoglu, S. H. Lin, and D. Tesar, Accessibility and controllability of flexible robotic manipulators, *ASME J. Dynamic Sys., Meas., Control,* **114** (1): 50–58, 1992.
9. J. J. Craig, *Introduction to Robotics: Mechanics and Control,* 2nd ed., Reading, MA: Addision-Wesley, 1989.

SABRI TOSUNOGLU
Florida International University

ROBOTIC REMOTE CONTROL. See TELEROBOTICS.