6th International Conference
"Research and Development in Mechanical Industry"
RaDMI 2006
13 - 17. September 2006, Budva, Montenegro

TRANSFORMATION MATRIX APPROACH TO DYNAMICS OF MANIPULATIVE ROBOTS

Ivan Milićević[1], Dragan Golubović[2], Nikola Slavković[3]
[1] Technical faculty, Čačak, SERBIA, E-mail: ivan_milicevic@beotel.yu, [2] E-mail: mehatron@ptt.yu
[3] Faculty of Mechanical Engineering, Belgrade, SERBIA, E-mail: rstanka@eunet.yu

Summary: *This paper considers the problem of defining and solving problems in robot mechanics that is fundamental for successful design and performance of robot control system. It considers a few different approaches for describing dynamics of robots and forming of dynamics models with purpose to get a set of mathematical equations describing activity of torques and forces on robot arm in form of motion equations describing the dynamic behavior of the manipulator. It describes the formulation, characteristics and properties of dynamics equations of motion, which are suitable for control purposes, through three different formulations, whereby advantages and disadvantages of each are described separately. To realize practical application of dynamics calculation and solving them using the PC computer, the method use 3×3 rotation matrices and 4×4 homogeneous transformation matrices.*

Key words: *robot dynamics, transformation matrix, equations of motion*

1. INTRODUCTION

Dynamics of robots deals with the mathematical formulations to describe activity of torques and forces on manipulator in form of dynamic equations of motion. The dynamic equations of motion of a manipulator are a set of mathematical equations describing the dynamic behavior of the manipulator. Such equations of motion are useful for computer simulation of the robot arm motion, the design of suitable control equations for a robot arm, and the evaluation of the kinematics design and structure of a robot arm. The purpose of manipulator control is to maintain the dynamic response of a computer-based manipulator in accordance with some prespecified system performance and desired goals. In general, the dynamic performance of a manipulator directly depends on the efficiency of the control algorithms and the dynamic model of the manipulator. The control problem consists of obtaining dynamic models of the physical robot arm system and then specifying corresponding control laws or strategies to achieve the desired system response and performance. This paper describes the formulation, characteristics, and properties of the dynamic equations of motion which are suitable for control purposes through three different formulations:

- Lagrange-Euler formulation,
- Newton-Euler formulation and,
- D'Alembert formulation.

2. LAGRANGE - EULER FORMULATION

The general motion equations of a manipulator can conveniently be expressed through the direct application of the Lagrange-Euler formulation to nonconservative systems. It use the Denavit-Hartenberg matrix representation [4] to describe the spatial displacement between the neighboring link coordinate frames to obtain the link kinematics information, and they employ the lagrangian dynamics technique to derive the dynamic equations of a manipulator. The direct application of the lagrangian dynamics formulation, together with the Denavit-Hartenberg link coordinate representation, results in a convenient and compact algorithmic description of the manipulator equations of motion. The algorithm is expressed by matrix operations and facilitates both analysis and computer implementation. The derivation of the dynamic equations of an n degrees of freedom manipulator is based on the understanding of:

1. The 4×4 homogeneous coordinate transformation matrix, ${}^{i-1}\mathbf{A}_i$ which describes the spatial relationship between the ith and the $(i-1)$th link coordinate frames. It relates a point fixed in link i expressed in homogeneous coordinates with respect to the ith coordinate system to the $(i-1)$th coordinate system.

$$ {}^{i-1}\mathbf{A}_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.1} $$

2. The Lagrange-Euler equation:

$$ \frac{d}{dt}\left[\frac{\partial L}{\partial \dot{q}_i}\right] - \frac{\partial L}{\partial q_i} = \tau_i \qquad i=1,2,...,n \tag{2.2} $$

where:

$L = E_K - E_P$ - Lagrangian function

E_K – total kinetic energy of the robot arm

E_P – total potential energy of the robot arm

q_i – generalized coordinates of the robot arm

$\dot{q}_i$ – first time derivative of the generalized coordinate q_i

τ_i – generalized force (or torque) applied to the system at joint i to drive link i.

From the above Lagrange-Euler equation, one is required to properly choose a set of generalized coordinates to describe the system. Generalized coordinates are used as a convenient set of coordinates which completely describe the location (position and orientation) of a system with respect to a reference coordinate frame. For a simple manipulator with rotary-prismatic joints, various sets of generalized coordinates are available to describe the manipulator. However, since the angular positions of the joints are readily available because they can be measured by potentiometers or encoders or other sensing devices, they provide a natural correspondence with the generalized coordinates. This, in effect, corresponds to the generalized coordinates with the joint variable defined in each of the 4×4 link coordinate transformation matrices. Hereafter, because of limited paper size, there are given only final formulas in function of dynamic coefficients, while is details derivation of the equations of motion of an n degrees of freedom manipulator, based on the homogeneous coordinate transformation matrices, described in [1].

Lagrangian function $L = E_K - E_P$ is given by:

$$ L = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{i}\sum_{k=1}^{i}\left[\mathrm{Tr}\left(\mathbf{U}_{ij}\mathbf{J}_i\mathbf{U}_{ik}^T\right)\dot{q}_j\dot{q}_k\right] + \sum_{i=1}^{n} m_i\mathbf{g}\left({}^0\mathbf{A}_i\,{}^i\bar{\mathbf{r}}_i\right) \tag{2.3} $$

where: $\mathbf{U}_{ij} \triangleq \dfrac{\partial\left({}^0\mathbf{A}_i\right)}{\partial q_j}$; $\mathbf{U}_{ijk} \triangleq \dfrac{\partial \mathbf{U}_{ij}}{\partial q_k}$.

Generalized torque (or force) τ_i for joint i actuator to drive the ith link of the manipulator:

$$ \tau_i = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = \sum_{j=i}^{n}\sum_{k=1}^{j}\mathrm{Tr}\left(\mathbf{U}_{jk}\mathbf{J}_j\mathbf{U}_{ji}^T\right)\ddot{q}_k + \sum_{j=i}^{n}\sum_{k=1}^{j}\sum_{m=1}^{j}\left[\mathrm{Tr}\left(\mathbf{U}_{jkm}\mathbf{J}_j\mathbf{U}_{ji}^T\right)\dot{q}_k\dot{q}_m\right] + \sum_{j=i}^{n} m_j\mathbf{g}\mathbf{U}_{ji}\,{}^j\bar{\mathbf{r}}_j \tag{2.4} $$

for i=1, 2,..., n

The above equation can be expressed in a much simpler matrix notation form as:

$$ \tau_i = \sum_{k=1}^{n} D_{ik}\ddot{q}_k + \sum_{k=1}^{n}\sum_{m=1}^{n} h_{ikm}\dot{q}_k\dot{q}_m + c_i \;;\quad i = 1, 2,..., n \tag{2.5} $$

or in a matrix form as:

$$ \boldsymbol{\tau}(t) = \mathbf{D}\left(\mathbf{q}(t)\right)\ddot{\mathbf{q}}(t) + \mathbf{h}\left(\mathbf{q}(t),\dot{\mathbf{q}}(t)\right) + \mathbf{c}\left(\mathbf{q}(t)\right) \tag{2.6} $$

where:

$\boldsymbol{\tau}(t)$ – n×1 generalized torque vector applied at joints $i=1,2,...,n;$ that is:

$$ \boldsymbol{\tau}(t) = [\tau_1(t), \tau_2(t),..., \tau_n(t)]^T \tag{2.7} $$

$\mathbf{q}(t)$ – n×1 vector of the joint variables of the robot arm and can be expressed as:

$$ \mathbf{q}(t) = [q_1(t), q_2(t),..., q_n(t)]^T \tag{2.8} $$

$\dot{\mathbf{q}}(t)$ – n×1 vector of the joint velocity of the robot am and can be expressed as:

$$ \dot{\mathbf{q}}(t) = [\dot{q}_1(t), \dot{q}_2(t),..., \dot{q}_n(t)]^T \tag{2.9} $$

$\ddot{\mathbf{q}}(t)$ – n×1 vector of the acceleration of the joint variables $\mathbf{q}(t)$ and can be expressed as:

$$\ddot{\mathbf{q}}(t) = [\ddot{q}_1(t), \ddot{q}_2(t), ..., \ddot{q}_n(t)]^T \quad \textbf{(2.10)}$$

$\mathbf{D}(\mathbf{q})$ – n×n inertial acceleration-related symmetric matrix whose elements are:

$$D_{ik} = \sum_{j=\max(i,k)}^{n} Tr\left(\mathbf{U}_{jk}\mathbf{J}_j\mathbf{U}_{ji}^T\right) \; ; \qquad i, k = 1, 2, ..., n \quad \textbf{(2.11)}$$

$\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})$ – n×1 nonlinear Coriolis and centrifugal force vector whose elements are:

$$\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = [h_1(t), h_2(t), ..., h_n(t)]^T \quad \textbf{(2.12)}$$

where:

$$h_i = \sum_{k=1}^{n}\sum_{m=1}^{n} h_{ikm}\dot{q}_k\dot{q}_m \; ; \qquad i = 1, 2, ..., n \quad \textbf{(2.13)}$$

$$h_{ikm} = \sum_{j=\max(i,k,m)}^{n} Tr\left(\mathbf{U}_{jkm}\mathbf{J}_j\mathbf{U}_{ji}^T\right) \; ; \qquad i, k, m = 1, 2, ..., n \quad \textbf{(2.14)}$$

$\mathbf{c}(\mathbf{q})$ – n×1 gravity loading force vector whose elements are:

$\mathbf{c}(\mathbf{q}) = [c_1(t), c_2(t), ..., c_n(t)]^T$

where:

$$c_i = \sum_{j=i}^{n}\left(-m_j \, \mathbf{g} \, \mathbf{U}_{ji} \, {}^j\bar{\mathbf{r}}_j\right) \; ; \qquad i = 1, 2, ..., n \quad \textbf{(2.15)}$$

The coefficients c_i, D_{ik} and h_{ikm} in Eqs. (2.11) to (2.15) are functions of both the joint variables and inertial parameters of the manipulator, and sometimes are called the *dynamic coefficients* of the manipulator. The physical meaning of these dynamic coefficients can easily be seen from the Lagrange-Euler equations of motion given by Eqs. (2.6) to (2.15):

1. The coefficient c_i represents the gravity loading terms due to the links and is defined by Eq. (2.15).
2. The coefficient D_{ik} is related to the acceleration of the joint variables and is defined by Eq. (2.11). In particular, for $i=k$, D_{ik} is related to the acceleration of joint i where the driving torque τ_i acts, while for $i\neq k$, D_{ik} is related to the reaction torque (or force) induced by the acceleration of joint k and acting at joint i, or vice versa. Since the inertia matrix is symmetric and $Tr(\mathbf{A}) = Tr(\mathbf{A}^T)$, it can be shown that $D_{ik} = D_{ki}$.
3. The coefficient h_{ikm} is related to the velocity of the joint variables and is defined by Eqs. (2.13) and (2.14). The last two indices, km, are related to the velocities of joints k and m, whose dynamic interplay induces a reaction torque (or force) at joint i. Thus, the first index i is always related to the joint where the velocity-induced reaction torques (or forces) are "felt". In particular, for $k=m$, h_{ikk} is related to the centrifugal force generated by the angular velocity of joint k and "felt" at joint i, while for $k\neq m$, h_{ikm} is related to the Coriolis force generated by the velocities of joints k and m and "felt" at joint i. It is noted that, for a given i, we have $h_{ikm} = h_{imk}$ which is apparent by physical reasoning.

In evaluating these coefficients, it is worth noting that some of the coefficients may be zero for the following reasons:

1. The particular kinematics design of a manipulator can eliminate some dynamic coupling (D_{ij} and h_{ikm} coefficients) between joint motions.
2. Some of the velocity-related dynamic coefficients have only a dummy existence in Eqs. (2.13) and (2.14); that is, they are physically nonexistent. (For instance, the centrifugal force will not interact with the motion of that joint which generates it, that is, $h_{iii}=0$ always; however, it can interact with motions at the other joints in the chain, that is, we can have $h_{jii} \neq 0$.)
3. Due to particular variations in the link configuration during motion, some dynamic coefficients may become zero at particular instants of time.

The motion equations of a manipulator as given by Eqs. (2.6) to (2.15) are coupled, nonlinear, second-order ordinary differential equations. These equations are in symbolic differential equation form and they include all inertial, centrifugal and Coriolis, and gravitational effects of the links. For a given set of applied torques τ_i $(i=1, 2, ..., n)$ as a function of time, Eq. (2.6) should be integrated simultaneously to obtain the actual motion of the manipulator in terms of the time history of the joint variables $\boldsymbol{q}(t)$. Then, the time history of the joint variables can be transformed to obtain the time history of the hand motion (hand trajectory) by using the appropriate homogeneous transformation matrices. Or, if the time history of the joint variables, the joint velocities, and the joint accelerations is known ahead of time from a trajectory planning program, then Eqs. (2.6) to (2.15) can be utilized to compute the applied torques $\tau(t)$ as a function of time which is required to produce the particular planned manipulator motion.

Because of its matrix structure, the L-E equations of motion are appealing from the closed-loop control viewpoint in that they give a set of state equations. This form allows design of a control law that easily compensates for all the nonlinear effects. Quite often in designing a feedback controller for a manipulator, the dynamic coefficients are used to minimize the nonlinear effects of the reaction forces.

It is of interest to evaluate the computational complexities inherent in obtaining the coefficients in Eqs. (2.11) to (2.15). Computationally, these equations of motion are extremely inefficient as compared with other formulations. In the next section are given the motion equations of a robot arm which will prove to be more efficient in computing the nominal torques.

3. NEWTON - EULER FORMULATION

As an alternative to deriving more efficient equations of motion, several investigators turned to Newton's second law and developed various forms of Newton-Euler equations of motion for an open kinematics chain. This formulation when applied to a robot arm results in a set of forward and backward recursive equations. The most significant aspect of this formulation is that the computation time of the applied torques can be reduced significantly to allow real-time control. The derivation is based on the D'Alembert principle and a set of mathematical equations that describe the kinematics relation of the moving links of a robot arm with respect to the base coordinate system. For defining the kinematics of manipulator, the method use 3×3 rotation matrices which transform any vector with reference to coordinate frame $(\boldsymbol{x_i}, \boldsymbol{y_i}, \boldsymbol{z_i})$ to the coordinate system $(\boldsymbol{x_{i-1}}, \boldsymbol{y_{i-1}}, \boldsymbol{z_{i-1}})$ and the position vector of the origin of the rotated coordinate system with respect to the reference system $\boldsymbol{p_i^*}$.

$$^{i-1}\mathbf{R}_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i \end{bmatrix}$$

3.1. Recursive Newton-Euler Equations of Motion for Manipulators

From the kinematics information of each link, we can describe the motion of the robot arm links by applying D'Alembert's principle to each link. D'Alembert's principle applies the conditions of static equilibrium to problems in dynamics by considering both the externally applied driving forces and the reaction forces of mechanical elements which resist motion. D'Alembert's principle applies for all instants of time. It is actually a slightly modified form of Newton's second law of motion, and can be stated as:
For any body, the algebraic sum of externally applied forces and the forces resisting motion in any given direction is zero.

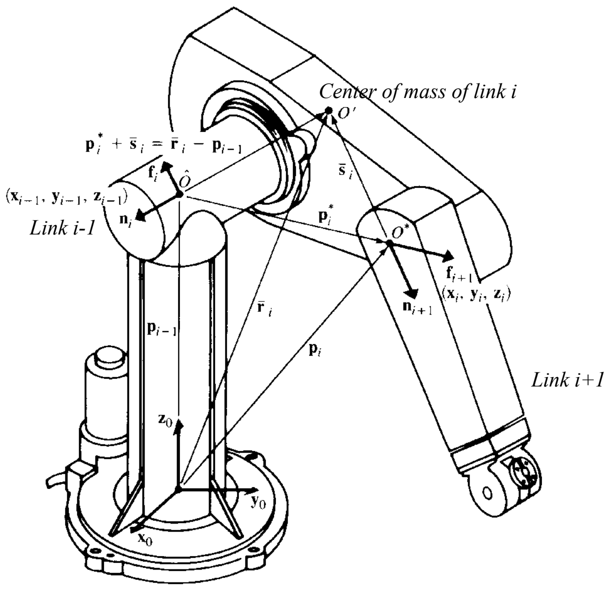

Figure 1: Forces and moments on link *i*

Consider a link *i* as shown in Fig. 1 and let the origin *O'* be situated at its center of mass. The variables defined in Fig. 1, expressed with respect to the base reference system $(\boldsymbol{x_o}, \boldsymbol{y_o}, \boldsymbol{z_o})$, are:

m_i – total mass of link *i*

$\bar{\mathbf{r}}_i$ – position of the center of mass of link *i* from the origin of the base reference frame

$\bar{\mathbf{s}}_i$ – position of the center of mass of link *i* from the origin of the coordinate system $(\boldsymbol{x_i}, \boldsymbol{y_i}, \boldsymbol{z_i})$

$\mathbf{p}_i^*$– the origin of the *i*th coordinate frame with respect to the *(i - 1)*th coordinate system

$\bar{\mathbf{v}}_i = \dfrac{d\bar{\mathbf{r}}_i}{dt}$ – linear velocity of the center of mass of link *i*

$\bar{\mathbf{a}}_i = \dfrac{d\bar{\mathbf{v}}_i}{dt}$ – linear acceleration of the center of mass of link *i*

$\mathbf{F}_i$ – total external force exerted on link *i* at the center of mass
$\mathbf{M}_i$ – total external moment exerted on link *i* at the center of mass
$\mathbf{I}_i$ – inertia matrix of link *i* about its center of mass with reference to the coordinate system $(\mathbf{x}_o, \mathbf{y}_o, \mathbf{z}_o)$
$\mathbf{f}_i$ – force exerted on link *i* by link *i - 1* at the coordinate frame $(\mathbf{x}_{i-1}, \mathbf{y}_{i-1}, \mathbf{z}_{i-1})$
to support link *i* and the links above it
$\mathbf{n}_i$ – moment exerted on link *i* by link *i - 1* at the coordinate frame $(\mathbf{x}_{i-1}, \mathbf{y}_{i-1}, \mathbf{z}_{i-1})$

In summary, the Newton-Euler equations of motion consist of a set of forward and backward recursive equations. They are listed in Table 1. For the forward recursive equations, linear velocity and acceleration, angular velocity and acceleration of each individual link, are propagated from the base reference system to the end-effector. For the backward recursive equations, the torques and forces exerted on each link are computed recursively from the end-effector to the base reference system. Hence, the forward equations propagate kinematics information of each link from the base reference frame to the hand, while the backward equations compute the necessary torques/forces for each joint from the hand to the base reference system.

Table 1: Recursive Newton-Euler equations of motion

Forward equations: *i = 1, 2, ..., n*

$$\boldsymbol{\omega}_i = \begin{cases} \boldsymbol{\omega}_{i-1} + \mathbf{z}_{i-1}\dot{q}_i & \text{if link } i \text{ is rotational} \\ \boldsymbol{\omega}_{i-1} & \text{if link } i \text{ is translational} \end{cases}$$

$$\dot{\boldsymbol{\omega}}_i = \begin{cases} \dot{\boldsymbol{\omega}}_{i-1} + \mathbf{z}_{i-1}\ddot{q}_i + \boldsymbol{\omega}_{i-1} \times (\mathbf{z}_{i-1}\dot{q}_i) & \text{if link } i \text{ is rotational} \\ \dot{\boldsymbol{\omega}}_{i-1} & \text{if link } i \text{ is translational} \end{cases}$$

$$\dot{\mathbf{v}}_i = \begin{cases} \dot{\boldsymbol{\omega}}_i \times \mathbf{p}_i^* + \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times \mathbf{p}_i^*) + \dot{\mathbf{v}}_{i-1} & \text{if link } i \text{ is rotational} \\ \mathbf{z}_{i-1}\ddot{q}_i + \dot{\boldsymbol{\omega}}_i \times \mathbf{p}_i^* + 2\boldsymbol{\omega}_i \times (\mathbf{z}_{i-1}\dot{q}_i) + \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times \mathbf{p}_i^*) + \dot{\mathbf{v}}_{i-1} & \text{if link } i \text{ is translational} \end{cases}$$

$$\bar{\mathbf{a}}_i = \dot{\boldsymbol{\omega}}_i \times \bar{\mathbf{s}}_i + \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times \bar{\mathbf{s}}_i) + \dot{\mathbf{v}}_i$$

Backward equations: *i = n, n-1, ..., 1*

$$\mathbf{F}_i = m_i \bar{\mathbf{a}}_i$$

$$\mathbf{M}_i = \mathbf{I}_i \dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times (\mathbf{I}_i \boldsymbol{\omega}_i)$$

$$\mathbf{f}_i = \mathbf{F}_i + \mathbf{f}_{i+1}$$

$$\mathbf{n}_i = \mathbf{n}_{i+1} + \mathbf{p}_i^* \times \mathbf{f}_{i+1} + (\mathbf{p}_i^* + \bar{\mathbf{s}}_i) \times \mathbf{F}_i + \mathbf{M}_i$$

$$\tau_i = \begin{cases} \mathbf{n}_i^T \mathbf{z}_{i-1} + b_i \dot{q}_i & \text{if link } i \text{ is rotational} \\ \mathbf{f}_i^T \mathbf{z}_{i-1} + b_i \dot{q}_i & \text{if link } i \text{ is translational} \end{cases}$$

where b_i is the viscous damping coefficient for joint *i*.

The "usual" initial conditions are: $\omega_0 = \dot{\omega}_0 = v_0 = 0$ i $\dot{\mathbf{v}}_0 = (g_x, g_y, g_z)^T$, where $|\mathbf{g}|$ =9.8062 m/s^2.

3.2. Recursive Equations of Motion of a Link about Its Own Coordinate Frame

The above equations of motion of a robot arm indicate that the resulting Newton-Euler dynamic equations are a set of compact forward and backward recursive equations. One obvious drawback of the above recursive equations of motion is that all the inertial matrices $\boldsymbol{I}_i$ and the physical geometric parameters ($\bar{\mathbf{r}}_i$, $\bar{\mathbf{s}}_i$, $\mathbf{p}_{i-1}$, $\mathbf{p}_i^*$) are referenced to the base coordinate system. As a result, they change as the robot arm is moving. Because of that, the above N-E equations of motion are improved by referencing all velocities, accelerations, inertial matrices, location of center of mass of each link, and forces/moments to their own link coordinate systems. Because of the nature of the formulation and the method of systematically computing the joint torques, computations are much simpler. The most important consequence of this modification is that the computation time of the applied torques is found linearly proportional to the number of joints of the robot arm and independent of the robot arm configuration. This enables the implementation of a simple real-time control algorithm for a robot arm in the joint-variable space. Hence, in summary, efficient Newton-Euler equations of motion are a set of forward and backward recursive equations with the dynamics and kinematics of each link referenced to its own coordinate system. A list of the recursive equations are found in Table 2.

Table 2: Efficient recursive Newton-Euler equations of motion

Forward equations: *i = 1, 2, …, n*

$$ {}^{i}\mathbf{R}_0\boldsymbol{\omega}_i = \begin{cases} {}^{i}\mathbf{R}_{i-1}\left({}^{i-1}\mathbf{R}_0\boldsymbol{\omega}_{i-1} + \mathbf{z}_0\dot{q}_i\right) & \text{if link } i \text{ is rotational} \\ {}^{i}\mathbf{R}_{i-1}\left({}^{i-1}\mathbf{R}_0\boldsymbol{\omega}_{i-1}\right) & \text{if link } i \text{ is translational} \end{cases} $$

$$ {}^{i}\mathbf{R}_0\dot{\boldsymbol{\omega}}_i = \begin{cases} {}^{i}\mathbf{R}_{i-1}\left[{}^{i-1}\mathbf{R}_0\dot{\boldsymbol{\omega}}_{i-1} + \mathbf{z}_0\ddot{q}_i + \left({}^{i-1}\mathbf{R}_0\boldsymbol{\omega}_{i-1}\right)\times\mathbf{z}_0\dot{q}_i\right] & \text{if link } i \text{ is rotational} \\ {}^{i}\mathbf{R}_{i-1}\left({}^{i-1}\mathbf{R}_0\dot{\boldsymbol{\omega}}_{i-1}\right) & \text{if link } i \text{ is translational} \end{cases} $$

$$ {}^{i}\mathbf{R}_0\dot{\mathbf{v}}_i = \begin{cases} \left({}^{i}\mathbf{R}_0\dot{\boldsymbol{\omega}}_i\right)\times\left({}^{i}\mathbf{R}_0\mathbf{p}_i^*\right) + \left({}^{i}\mathbf{R}_0\boldsymbol{\omega}_i\right)\times\left[\left({}^{i}\mathbf{R}_0\boldsymbol{\omega}_i\right)\times\left({}^{i}\mathbf{R}_0\mathbf{p}_i^*\right)\right] + {}^{i}\mathbf{R}_{i-1}\left({}^{i-1}\mathbf{R}_0\dot{\mathbf{v}}_{i-1}\right) \\ {}^{i}\mathbf{R}_{i-1}\left(\mathbf{z}_0\ddot{q}_i + {}^{i-1}\mathbf{R}_0\dot{\mathbf{v}}_{i-1}\right) + \left({}^{i}\mathbf{R}_0\dot{\boldsymbol{\omega}}_i\right)\times\left({}^{i}\mathbf{R}_0\mathbf{p}_i^*\right) + 2\left({}^{i}\mathbf{R}_0\boldsymbol{\omega}_i\right)\times\left({}^{i}\mathbf{R}_{i-1}\mathbf{z}_0\dot{q}_i\right) + \left({}^{i}\mathbf{R}_0\boldsymbol{\omega}_i\right)\times\left[\left({}^{i}\mathbf{R}_0\boldsymbol{\omega}_i\right)\times\left({}^{i}\mathbf{R}_0\mathbf{p}_i^*\right)\right] \end{cases} $$

$$ {}^{i}\mathbf{R}_0\bar{\mathbf{a}}_i = \left({}^{i}\mathbf{R}_0\dot{\boldsymbol{\omega}}_i\right)\times\left({}^{i}\mathbf{R}_0\bar{\mathbf{s}}_i\right) + \left({}^{i}\mathbf{R}_0\boldsymbol{\omega}_i\right)\times\left[\left({}^{i}\mathbf{R}_0\boldsymbol{\omega}_i\right)\times\left({}^{i}\mathbf{R}_0\bar{\mathbf{s}}_i\right)\right] + {}^{i}\mathbf{R}_0\dot{\mathbf{v}}_i $$

Backward equations: *i = n, n-1, …, 1*

$$ {}^{i}\mathbf{R}_0\mathbf{f}_i = {}^{i}\mathbf{R}_{i+1}\left({}^{i+1}\mathbf{R}_0\mathbf{f}_{i+1}\right) + m_i\,{}^{i}\mathbf{R}_0\bar{\mathbf{a}}_i $$

$$ {}^{i}\mathbf{R}_0\mathbf{n}_i = {}^{i}\mathbf{R}_{i+1}\left[\left({}^{i+1}\mathbf{R}_0\mathbf{n}_{i+1}\right) + \left({}^{i+1}\mathbf{R}_0\mathbf{p}_i^*\right)\times\left({}^{i+1}\mathbf{R}_0\mathbf{f}_{i+1}\right)\right] + \left({}^{i}\mathbf{R}_0\mathbf{p}_i^* + {}^{i}\mathbf{R}_0\bar{\mathbf{s}}_i\right)\times\left(m_i\,{}^{i}\mathbf{R}_0\bar{\mathbf{a}}_i\right) + $$
$$ + \left({}^{i}\mathbf{R}_0\mathbf{I}_i\,{}^{0}\mathbf{R}_i\right)\left({}^{i}\mathbf{R}_0\dot{\boldsymbol{\omega}}_i\right) + \left({}^{i}\mathbf{R}_0\boldsymbol{\omega}_i\right)\times\left[\left({}^{i}\mathbf{R}_0\mathbf{I}_i\,{}^{0}\mathbf{R}_i\right)\left({}^{i}\mathbf{R}_0\boldsymbol{\omega}_i\right)\right] $$

$$ \tau_i = \begin{cases} \left({}^{i}\mathbf{R}_0\mathbf{n}_i\right)^T\left({}^{i}\mathbf{R}_{i-1}\mathbf{z}_0\right) + b_i\dot{q}_i & \text{if link } i \text{ is rotational} \\ \left({}^{i}\mathbf{R}_0\mathbf{f}_i\right)^T\left({}^{i}\mathbf{R}_{i-1}\mathbf{z}_0\right) + b_i\dot{q}_i & \text{if link } i \text{ is translational} \end{cases} $$

where $\mathbf{z}_0=(0, 0, 1)^T$, and b_i is the viscous damping coefficient for joint i.

The "usual" initial conditions are: $\boldsymbol{\omega}_0 = \dot{\boldsymbol{\omega}}_0 = \mathbf{v}_0 = \mathbf{0}$ i $\dot{\mathbf{v}}_0 = \left(g_x, g_y, g_z\right)^T$, where $|\mathbf{g}|$=9,8062 m/s^2.

4. GENERALIZED D'ALEMBERT EQUATIONS OF MOTION

In order to obtain an efficient set of closed-form equations of motion, one can utilize the relative position vector and rotation matrix representation to describe the kinematics information of each link, obtain the kinetic and potential energies of the robot arm to form the lagrangian function, and apply the Lagrange-Euler formulation to obtain the equations of motion. In this section is describe a Lagrange form of D'Alembert equations of motion or generalized D'Alembert equations of motion. Focus is on robot arms with rotary joints.

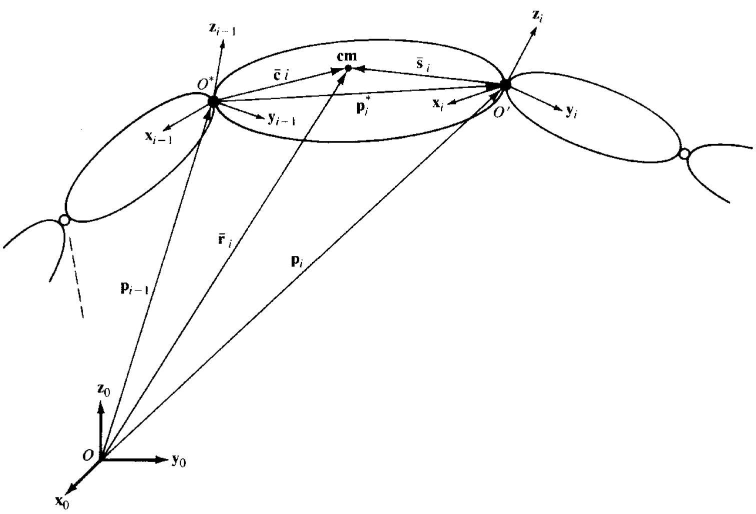

Figure 2: Vector definition in the generalized D'Alembert equations

Equation for the generalized applied torque exerted at joint i to drive link i:

$$\sum_{j=1}^{n} D_{ij}\ddot{\theta}_j(t) + h_i^{tran}(\theta,\dot{\theta}) + h_i^{rot}(\theta,\dot{\theta}) + c_i = \tau_i(t) \tag{4.1}$$

where, for $i = 1, 2,\ldots, n,$

$$\begin{aligned} D_{ij} &= D_{ij}^{rot} + D_{ij}^{tran} = \sum_{s=j}^{n}\left[\left({}^{s}\mathbf{R}_0\mathbf{z}_{i-1}\right)^T \mathbf{I}_s \left({}^{s}\mathbf{R}_0\mathbf{z}_{j-1}\right)\right] + \sum_{s=j}^{n}\left\{ m_s\left[\mathbf{z}_{j-1}\times\left(\sum_{k=j}^{s-1}\mathbf{p}_k^* + \bar{\mathbf{c}}_s\right)\right]\cdot\left[\mathbf{z}_{i-1}\times\left(\bar{\mathbf{r}}_s - \mathbf{p}_{i-1}\right)\right]\right\} = \\ &= \sum_{s=j}^{n}\left[\left({}^{s}\mathbf{R}_0\mathbf{z}_{i-1}\right)^T \mathbf{I}_s \left({}^{s}\mathbf{R}_0\mathbf{z}_{j-1}\right)\right] + \sum_{s=j}^{n}\left\{ m_s\left[\mathbf{z}_{j-1}\times\left(\bar{\mathbf{r}}_s - \mathbf{p}_{j-1}\right)\right]\cdot\left[\mathbf{z}_{i-1}\times\left(\bar{\mathbf{r}}_s - \mathbf{p}_{i-1}\right)\right]\right\}, \qquad i \le j \end{aligned} \tag{4.2}$$

$$\begin{aligned} h_i^{tran}\left(\theta,\dot{\theta}\right) &= \sum_{s=i}^{n}\left\{ m_s\left\{\sum_{k=1}^{s-1}\left\{\left[\left(\sum_{p=1}^{k}\dot{\theta}_p\mathbf{z}_{p-1}\right)\times\left[\left(\sum_{q=1}^{k}\dot{\theta}_q\mathbf{z}_{q-1}\right)\times\mathbf{p}_k^*\right]\right] + \right.\right.\right. \\ &\left.\left.\left. + \left[\sum_{p=2}^{k}\left[\left(\sum_{q=1}^{p-1}\dot{\theta}_q\mathbf{z}_{q-1}\right)\times\dot{\theta}_p\mathbf{z}_{p-1}\right]\times\mathbf{p}_k^*\right]\right\}\right\}\cdot\left[\mathbf{z}_{i-1}\times\left(\bar{\mathbf{r}}_s - \mathbf{p}_{i-1}\right)\right]\right\} + \\ &+ \sum_{s=i}^{n}\left\{ m_s\left\{\left[\left(\sum_{p=1}^{s}\dot{\theta}_p\mathbf{z}_{p-1}\right)\times\left[\left(\sum_{q=1}^{s}\dot{\theta}_q\mathbf{z}_{q-1}\right)\times\bar{\mathbf{c}}_s\right]\right] + \left[\sum_{p=2}^{s}\left[\left(\sum_{q=1}^{p-1}\dot{\theta}_q\mathbf{z}_{q-1}\right)\times\dot{\theta}_p\mathbf{z}_{p-1}\right]\times\bar{\mathbf{c}}_s\right]\right\}\cdot\left[\mathbf{z}_{i-1}\times\left(\bar{\mathbf{r}}_s - \mathbf{p}_{i-1}\right)\right]\right\} \end{aligned} \tag{4.3}$$

$$\begin{aligned} h_i^{rot}\left(\theta,\dot{\theta}\right) &= \sum_{s=i}^{n}\left\{\left({}^{s}\mathbf{R}_0\mathbf{z}_{i-1}\right)^T \mathbf{I}_s\left\{\sum_{j=1}^{s}\left[\dot{\theta}_j\,{}^{s}\mathbf{R}_0\mathbf{z}_{j-1}\times\left(\sum_{k=j+1}^{s}\dot{\theta}_k\,{}^{s}\mathbf{R}_0\mathbf{z}_{k-1}\right)\right]\right\} + \right. \\ &\left. + \left[{}^{s}\mathbf{R}_0\mathbf{z}_{i-1}\times\left(\sum_{p=1}^{s}\dot{\theta}_p\,{}^{s}\mathbf{R}_0\mathbf{z}_{p-1}\right)\right]^T \mathbf{I}_s\left(\sum_{q=1}^{s}\dot{\theta}_q\,{}^{s}\mathbf{R}_0\mathbf{z}_{q-1}\right)\right\} \end{aligned} \tag{4.4}$$

and:

$$c_i = -\mathbf{g}\cdot\left[\mathbf{z}_{i-1}\times\sum_{j=i}^{n} m_j\left(\bar{\mathbf{r}}_j - \mathbf{p}_{i-1}\right)\right] \tag{4.5}$$

The dynamic coefficients D_{ij} and c_i are functions of both the joint variables and inertial parameters of the manipulator, while the h_i^{tran} and h_i^{rot} are functions of the joint variables, the joint velocities and inertial parameters of the manipulator. These coefficients have the following physical interpretation:

- The elements of the D_{ij} matrix are related to the inertia of the links in the manipulator. Equation (4.2) reveals the acceleration effects of joint j acting on joint i where the driving torque τ_i acts. The first term of Eq. (4.2) indicates the inertial effects of moving link j on joint i due to the rotational motion of link j, and vice versa. If $i=j$, it is the effective inertias felt at joint i due to the rotational motion of link i; while if $i \neq j$, it is the pseudo products of inertia of link j felt at joint i due to the rotational motion of link j. The second term has the same physical meaning except that it is due to the translational motion of link j acting on joint i.
- The $h_i^{tran}(q,\dot{q})$ term is related to the velocities of the joint variables. Equation (4.3) represents the combined centrifugal and Coriolis reaction torques felt at joint i due to the velocities of joints p and q resulting from the *translational* motion of links p and q. The first and third terms of Eq. (4.3) constitute, respectively, the centrifugal and Coriolis reaction forces from all the links below link s and link s itself, due to the translational motion of the links. If $p=q$, then it represents the centrifugal reaction forces felt at joint i. If $p \neq q$, then it indicates the Coriolis forces acting on joint i. The second and fourth terms of Eq. (4.3) indicate, respectively, the Coriolis reaction forces contributed from the links below link s and link s itself, due to the translational motion of the links.
- The $h_i^{rot}(q,\dot{q})$ term is also related to the velocities of the joint variables. Similar to Eq. (4.3), Eq. (4.4) reveals the combined centrifugal and Coriolis reaction torques felt at joint i due to the velocities of joints p and q resulting from the *rotational* motion of links p and q. The first term of Eq. (4.4) indicates purely the Coriolis reaction forces of joints p and q acting on joint i due to the rotational motion of the links. The second term is the combined centrifugal and Coriolis reaction forces acting on joint i. If $p=q$, then it indicates the centrifugal reaction forces felt at joint i, but if $p \neq q$, then it represents the Coriolis forces acting on joint i due to the rotational motion of the links.
- The coefficient c_i represents the gravity effects acting on joint i from the links above joint i.

At first sight, Eqs. (4.2) to (4.5) would seem to require a large amount of computation. However, most of the cross-product terms can be computed very fast. Table 3 summarizes the computational complexities of the Lagrange-Euler, Newton-Euler, and Generalized D'Alembert equations of motion in terms of required mathematical operations per trajectory set point.

Table 3: Comparison of robot arm dynamics computational complexities [3]

Approach	Lagrange-Euler	Newton-Euler	Generalized D'Alembert
Multiplications	$128/3n^4+512/3n^3+$ $+739/3n^2+160/3n$	132n	$13/6n^3+105/2n^2+$ $+268/3n+69$
Additions	$98/3n^4+781/6n^3+$ $+559/3n^2+245/6n$	111n-4	$4/3n^3+44n^2+$ $+146/3n+45$
Kinematics representation	4×4 Homogeneous transformation matrices	3×3 Rotation matrices and position vectors	3×3 Rotation matrices and position vectors
Equations of motion	Closed-form differential equations	Recursive equations	Closed-form differential equations

n - number of degrees of freedom of the robot arm.

5. CONCLUSION

Three different formulations for robot arm dynamics have been presented and discussed. The Lagrange-Euler equations of motion can be expressed in a well structured form (the interaction and coupling reaction forces in the equations should be easily identified so that an appropriate controller can be designed to compensate for their effects), but they are computationally difficult to utilize for real-time control purposes unless they are simplified. The Newton-Euler formulation results in a very efficient set of recursive equations, but they are difficult to use for deriving advanced control laws. The Generalized D'Alembert equations of motion give fairly well "structured" equations at the expense of higher computational cost. In addition to having faster computation time than the Lagrange-Euler equations of motion, the D'Alembert equations of motion explicitly indicate the contributions of the translational and rotational effects of the links. Such information is useful for control analysis in obtaining an appropriate approximate model of a manipulator. Furthermore, the D'Alembert equations of motion can be used in manipulator design. To briefly summarize the results, a user is able to choose between a formulation which is highly structured but computationally inefficient (Lagrange-Euler), a formulation which has efficient computations at the expense of the "structure" of the equations of motion (Newton-Euler), and a formulation which retains the "structure" of the problem with only a moderate computational penalty (Generalized D'Alembert).

REFERENCES

[1] MILIĆEVIĆ, I.: *Primena matrica transformacija pri rešavanju kinematike i dinamike manipulacionih robota*, Magistarski rad, Čačak, 2006.

[2] GOLUBOVIĆ, D., MIKIĆ, D., MILIĆEVIĆ, I.: *Kinematika robota sa korišćenjem matrica rotacionih transformacija*, IRMES '04, Istraživanje i razvoj mašinskih elemenata i sistema, Zbornik radova, str. 147-153, Kragujevac, 2004.

[3] FU, K. S., GONZALEZ, R. C., LEE, C. S. G.: *Robotics: Control, Sensing, Vision, And Intelligence*, Mc Graw-Hill Book Company, ISBN 0-07-100421-1, 1987.

[4] DENAVIT, J., HARTENBERG, R. S.: *A Kinematic Notation for Lower-Pair Mechanisms Based on Matricies*, Trans. of ASME J.App. Mech. vol. 77, pp. 215-221, June 1955.

[5] ROBOTIC AND AUTOMATION HANDBOOK, Edited By Thomas R. Kurfess, CRC Press, 2005.

[6] VUKOBRATOVIĆ, M.: Primenjena dinamika manipulacionih robota, NIP Tehnička knjiga, Beograd, 1990.