[1]Ahmet SHALA, [2]Ramë LIKAJ

DYNAMIC MODELING OF 3 DoF ROBOT MANIPULATOR

[1-2] University of Prishtina / Faculty of Mechanical Engineering, Prishtina, KOSOVO

Abstract: Dynamical Modeling of robots is commonly first important step of Modeling, Analysis and Control of robotic systems. This paper is focused on using Denavit-Hartenberg (DH) convention for kinematics and Newton-Euler Formulations for dynamic modeling of 3 DoF - Degree of Freedom of 3D robot. The process of deriving of dynamical model is done using Software Maple. Derived Dynamical Model of 3 Dof robot is converted for Matlab software use for future analysis, control and simulations.
Keywords: Modelling, dynamics, robot, analysis

INTRODUCTION

Dynamics is a huge field of study devoted to studying the forces required to cause motion. The dynamic motion of the manipulator arm in a robotic system is produced by the torques generated by the actuators. This relationship between the input torques and the time rates of change of the robot arm components configurations, represent the dynamic modeling of the robotic system which is concerned with the derivation of the equations of motion of the manipulator as a function of the forces and moments acting on. So, the dynamic modeling of a robot manipulator consists of finding the mapping between the forces exerted on the structures and the joint positions, velocities and accelerations. A good model has to satisfy two conflicting objectives.

A robot manipulator is basically a positioning device. To control the position we must know the dynamic properties of the manipulator in order to know how much force to exert on it to cause it to move: too little force and the manipulator is slow to react; too much force and the arm may crash into objects or oscillate about its desired position.

Deriving the dynamic equations of motion for robots is not a simple task due to the large number of degrees of freedom and nonlinearities present in the system. This part is concerned with the development of the dynamic model for 3 Dof robot and their kinematics and dynamics equations.

ROBOT STRUCTURE OF 3 DOF AND COORDINATE SYSTEMS

Based on structure of 3 Dof robot (Figure 1), is created Table 1 of Denavit-Hartenberg parameters for 3 DoF robot.

Table 1. Denavit-Hartenberg parameters for 3 DoF robot

Link #	a_i	α_i	d_i	θ_i
1	0	0	d_1	q_1^*
2	0	$-\beta$	d_2^*	0
3	0	0	L_{3a}	q_3^*

* Joint variable

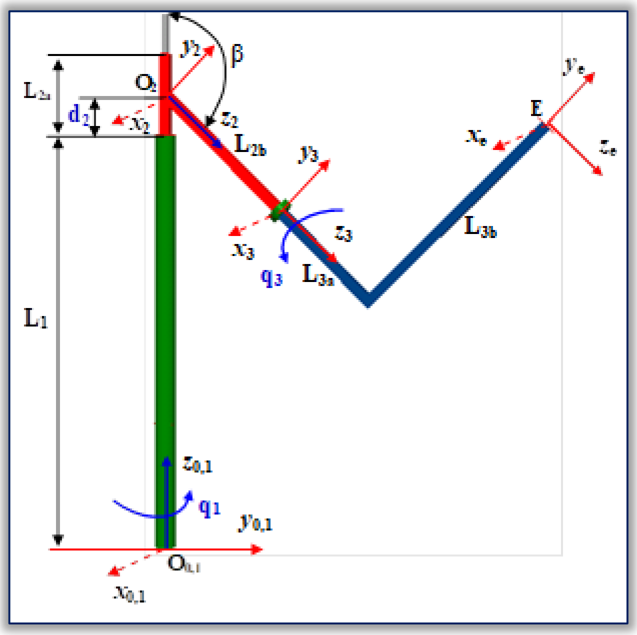

Figure 1. Symbolic representation - Axes rotations for Denavit-Hartenberg parameters

Denavit-Hartenberg transformation matrix for adjacent coordinate frames, i and i – 1.

$$A_i = \begin{bmatrix} \cos(\theta_i) & -\cos(\alpha_i)\cdot\sin(\theta_i) & \sin(\alpha_i)\cdot\sin(\theta_i) & a_i\cdot\cos(\theta_i) \\ \sin(\theta_i) & \cos(\alpha_i)\cdot\cos(\theta_i) & -\sin(\alpha_i)\cdot\cos(\theta_i) & a_i\cdot\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Orthogonal rotation matrix R_i which transforms a vector in the i-th coordinate frame to a coordinate frame which is parallel to the (i-1)-th coordinate frame is first 3x3 sub-matrices of A_i:

$$R_i = \begin{bmatrix} \cos(\theta_i) & -\cos(\alpha_i)\cdot\sin(\theta_i) & \sin(\alpha_i)\cdot\sin(\theta_i) \\ \sin(\theta_i) & \cos(\alpha_i)\cdot\cos(\theta_i) & -\sin(\alpha_i)\cdot\cos(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix}$$

for i=1,2, … , N, where $R_{N+1} = E = \mathrm{diag}(1)$

DYNAMIC EQUATIONS – NEWTON-EULER FORMULATION

Dynamics of robot is the study of motion with regard to forces (the study of the relationship between forces/torques and motion). A dynamic analysis of a manipulator is useful for the following purposes:

- It determines the joint forces and torques required to produce specified end-effector motions (the direct dynamic problem).
- It produces a mathematical model which simulates the motion of the manipulator under various loading conditions (the inverse dynamic problem) and/or control schemes.
- It provides a dynamic model for use in the control of the actual manipulator.

Dynamic modelling of mechanical structures can be a complex problem. In robotics, more specifically, in manipulators, there are two methodologies used for dynamic modelling.

NEWTON-EULER FORMULATION

The Newton-Euler formulation [1] shown in equations (1)-(9) computes the inverse dynamics (ie., joint torques/forces from joint positions, velocities, and accelerations) bases on two sets of recursions: the forward and backward recursions. The forward recursions (1)-(3) transform the kinematics variables from the base to the end-effector. The initial conditions (for i=0) assume that the manipulator is at rest in the gravitational field. The backward recursions (4)-(9) transform the forces and moments from the end-effector to the base, and culminate with the calculation of the joint torques/forces.

Angular velocity of the i-th coordinate frame

$$\omega_{i+1} = \begin{cases} R^T_{i+1} \cdot \left[\omega_i + z_0 \cdot \dot{\theta}_{i+1} \right] & \text{if joint is rotational} \\ R^T_{i+1} \cdot \omega_i & \text{if joint is translational} \end{cases} \tag{1}$$

where: $z_0 = [0 \quad 0 \quad 1]^T$

Angular acceleration of the i-th coordinate frame

$$\dot{\omega}_{i+1} = \begin{cases} R^T_{i+1} \cdot \left[\dot{\omega}_i + z_0 \cdot \ddot{\theta}_{i+1} + \omega_i \times (z_0 \cdot \dot{\theta}_{i+1}) \right] \\ \text{if joint is rotational} \\ R^T_{i+1} \cdot \dot{\omega}_i \\ \text{if joint is translational} \end{cases} \tag{2}$$

Linear acceleration of the i-th coordinate frame

$$\dot{v}_{i+1} = \begin{cases} R^T_{i+1} \cdot \dot{v}_i + \dot{\omega}_{i+1} \times p_{i+1} + \omega_{i+1} \times (\omega_{i+1} \times p_{i+1}) \\ \text{if joint is rotational} \\ R^T_{i+1} \left[\dot{v}_i + z_0 \ddot{d}_{i+1} + 2\omega_i \, x(z_0 \dot{d}_{i+1}) \right] + \\ + \dot{\omega}_{i+1} \, xp_{i+1} + \omega_{i+1} \, x(\omega_{i+1} \, xp_{i+1}) \\ \text{if joint is translational} \end{cases} \tag{3}$$

where: $p_i = [a_i \quad d_i \cdot \sin(\alpha_i) \quad d_i \cdot \sin(\alpha_i)]^T$ is position of the i-th coordinate frame with respect to the (i – 1)-th coordinate frame.

Initial conditions: $\omega_0 = \dot{\omega}_0 = v_0 = 0$; Gravitational acceleration: $\dot{v}_0 = [g_x \quad g_y \quad g_z]^T$.

Linear acceleration of the center-of-mass of link i

$$a_i = \dot{\omega}_i \times s_i + \omega_i \times (\omega_i \times s_i) + \dot{v}_i \tag{4}$$

where: s_i is position of center-of-mass of link i

Net force exerted on link i:

$$F_i = m_i \cdot a_i \tag{5}$$

Net moment exerted on link i

$$N_i = I_i \dot{\omega}_i + \omega_i \times (I_i \cdot \omega_i) \tag{6}$$

where $I_i = \begin{bmatrix} I_{ixx} & 0 & 0 \\ 0 & I_{iyy} & 0 \\ 0 & 0 & I_{izz} \end{bmatrix}$

I_i is moment of inertia tensor of link i about the centre-of-mass of link i (parallel to the i-th coordinate frame), with only principal inertias I_{ixx}, I_{iyy} and I_{izz}. Because of symmetry of link frames, cross-inertias can be used zero.

Force exerted on link i by link i – 1:

$$f_i = R^T_{i+1} \cdot f_{i+1} + F_i \tag{7}$$

Moment exerted on link i by link i – 1

$$n_i = R^T_{i+1} \cdot n_{i+1} + p_i \times f_i + N_i + s_i \times F_i \tag{8}$$

Joint torque/force at joint i:

$$\tau_i = \begin{cases} n_i^T \cdot (R^T_{i+1} \cdot z_0) & \text{if joint is rotational} \\ f_i^T \cdot (R^T_{i+1} \cdot z_0) & \text{if joint is translational} \end{cases} \tag{9}$$

DERIVING OF DYNAMICAL MODEL FOR 3 DOF ROBOT

Based on Newton-Euler formulation (1-9), rotation matrices for links of robot (i=1,2,3) are:

$$R_1 = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 \\ \sin(q_1) & \cos(q_1) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\beta) & \sin(\beta) \\ 0 & -\sin(\beta) & \cos(\beta) \end{bmatrix}$$

$$R_3 = \begin{bmatrix} \cos(q_3) & -\sin(q_3) & 0 \\ \sin(q_3) & \cos(q_3) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Initial conditions are:

$$z_0 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \ \omega_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \ \dot{\omega}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \ v_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \ \dot{v}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Forward recursions, for 3 DoF robot, (i=1,2,3).
Angular velocity of the i-th coordinate frame:

$$\omega_1 = R_1{}^T \cdot (\omega_0 + z_0 \cdot \dot{q}_1)$$
$$\omega_2 = R_2{}^T \cdot \omega_1$$
$$\omega_3 = R_3{}^T \cdot (\omega_2 + z_0 \cdot \dot{q}_3)$$

Angular acceleration of the i-th coordinate frame:

$$\dot{\omega}_1 = R_1{}^T \cdot (\dot{\omega}_0 + z_0 \cdot \ddot{q}_1) + \omega_0 \times z_0 \cdot \dot{q}_1$$
$$\dot{\omega}_2 = R_2{}^T \cdot \dot{\omega}_1$$
$$\dot{\omega}_3 = R_3{}^T \cdot (\dot{\omega}_2 + z_0 \cdot \ddot{q}_3) + \omega_2 \times z_0 \cdot \dot{q}_3$$

Position of the i-th coordinate frame with respect to the (i-1)-th coordinate frame:

$$p_1 = \begin{bmatrix} 0 \\ 0 \\ L_1 \end{bmatrix}, \ p_2 = \begin{bmatrix} 0 \\ 0 \\ d_2 \end{bmatrix}, \ p_3 = \begin{bmatrix} 0 \\ 0 \\ L_{2b} \end{bmatrix}$$

Linear acceleration of the i-th coordinate frame:

$$\dot{v}_1 = R_1{}^T \cdot \dot{v}_0 + \dot{\omega}_1 \times p_1 + \dot{\omega}_1 \times (\dot{\omega}_1 \times p_1)$$
$$\dot{v}_2 = R_2{}^T \cdot \dot{v}_1 + z_0 \cdot \dot{q}_2 + (2\dot{\omega}_1 \times z_0 \cdot \dot{q}_2) + (\omega_1 \times p_1)$$
$$\dot{v}_3 = R_3{}^T \cdot \dot{v}_2 + \dot{\omega}_3 \times p_3 + \dot{\omega}_3 \times (\dot{\omega}_3 \times p_3)$$

Linear acceleration of the center-of-mass of link i:

$$\dot{a}_1 = \dot{\omega}_1 \times s_1 + \omega_1 \times (\omega_1 \times s_1) + \dot{v}_1$$
$$\dot{a}_2 = \dot{\omega}_2 \times s_2 + \omega_2 \times (\omega_2 \times s_2) + \dot{v}_2$$
$$\dot{a}_3 = \dot{\omega}_3 \times s_3 + \omega_3 \times (\omega_3 \times s_3) + \dot{v}_3$$

Moment of inertia tensor of link i about the center of mass of link i (parallel to the i-th coordinate frame), with only principal inertias I_{ixx}, I_{iyy} and I_{izz}. Because of symmetry of link frames, cross-inertias are used zero.

$$I_1 = \begin{bmatrix} I_{1xx} & 0 & 0 \\ 0 & I_{1yy} & 0 \\ 0 & 0 & I_{1zz} \end{bmatrix}, \ I_2 = \begin{bmatrix} I_{2xx} & 0 & 0 \\ 0 & I_{2yy} & 0 \\ 0 & 0 & I_{2zz} \end{bmatrix},$$

$$I_3 = \begin{bmatrix} I_{3xx} & 0 & 0 \\ 0 & I_{3yy} & 0 \\ 0 & 0 & I_{3zz} \end{bmatrix}$$

Net force and moment exerted on link i = 1,2,3:

$$F_1 = m_1 \cdot \dot{a}_1$$
$$F_2 = m_2 \cdot \dot{a}_2$$
$$F_3 = m_3 \cdot \dot{a}_3$$
$$N_1 = I_1 \cdot \dot{\omega}_1 + (\dot{\omega}_1 \times I_1 \cdot \dot{\omega}_1)$$
$$N_2 = I_2 \cdot \dot{\omega}_2 + (\dot{\omega}_2 \times I_2 \cdot \dot{\omega}_2)$$
$$N_3 = I_3 \cdot \dot{\omega}_3 + (\dot{\omega}_3 \times I_3 \cdot \dot{\omega}_3)$$

Backward recursion (i=3,2,1); Force and moment exerted on link i by link i-1.
By supposing that there are no outside load for end-effector are:

$$f_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \ n_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \ R_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

For link #3:

$$f_3 = R_4 \cdot f_4 + F_3$$
$$n_3 = R_4 \cdot n_4 + p_3 \times f_3 + N_3 + s_3 \times F_3$$

Torque at joint 3:

$$\tau_3 = n_3{}^T \cdot (R_3 \cdot z_0)$$

For link #2:

$$f_2 = R_3 \cdot f_3 + F_2$$
$$n_2 = R_3 \cdot n_3 + p_2 \times f_2 + N_2 + s_2 \times F_2$$

Force at joint 2:

$$\tau_2 = f_2{}^T \cdot (R_2 \cdot z_0)$$

For link #1:

$$f_1 = R_2 \cdot f_2 + F_1$$
$$n_1 = R_2 \cdot n_2 + p_1 \times f_1 + N_1 + s_1 \times F_1$$

Torque at joint 1:

$$\tau_1 = n_1{}^T \cdot (R_1 \cdot z_0)$$

In the end the vector of Forces-Torques for 3 DoF robot is:

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix}$$

Modeling-Calculations of 3 DoF robot is done using Maple software, equations are converted for Matlab use.

CONCLUSION

Based on presented paper can be concluded that Newton-Euler formulation is very useful for dynamical modeling of systems generally and robotic systems especially.
Use of Maple software is very useful for modelling of complex systems and representations of results symbolically –

representation of expressions of dynamical model with many characters (until 100000).

Opportunity of Maple software to convert derived expressions for Matlab use is very helpful for future analyses and simulations of systems.

Acknowledgements

The first author is profoundly thankful to the corresponding author Rame Likaj (email: rame.likaj@uni-pr.edu) which has paid attention to fulfill all requirements about this research work.

References

[1] Luh, J. Y. S., Walker, M. W. and Paul, R. P.; On-Line Computational Scheme for Mechanical Manipulators. Journal of Dynamic Systems, Measurement, and Control 102(2):69-76, June, 1980.
[2] Brady, M., et al. (editors). Robot Motion: Planning and Control. MIT Press, Cambridge, MA, 1982.
[3] Khosla, Pradeep, "Estimation of Robot Dynamics Parameters: Theory and Application". Institute for Software Research. Paper 651. http://repository.cmu.edu/isr/651, Carnegie Mellon University, 1987.
[4] A. Shala, R. Likaj; Design of Genetic Algorithm for optimization of Fuzzy Neural Network Controller, 8th International Conference Modern Technologies in Manufacturing, Cluj-Napoca, Romania, 2007.
[5] H.J. Sommer: Vector loops facilitate simple kinematic analysis of planar mechanisms, either closed-chain or open-chain, Mechanical & Nuclear Engineering Faculty, USA, www.mne.psu.edu/sommer/me50/
[6] J. M. McCarthy and G. S. Soh, 2010, Geometric Design of Linkages, Springer, New York.
[7] B. Paul, Kinematics and Dynamics of Planar Machinery, Prentice-Hall, NJ, 1979
[8] L. W. Tsai, Robot Analysis: The mechanics of serial and parallel manipulators, John-Wiley, NY, 1999.
[9] K. J. Waldron and G. L. Kinzel, Kinematics and Dynamics, and Design of Machinery, 2nd Ed., John Wiley and Sons, 2004.
[10] Torby, Bruce (1984). "Energy Methods". Advanced Dynamics for Engineers. HRW Series in Mechanical Engineering. United States of America: CBS College Publishing. ISBN 0-03-063366-4.

ACTA TECHNICA CORVINIENSIS BULLETIN OF ENGINEERING
ISSN: 2067-3809
copyright © University POLITEHNICA Timisoara,
Faculty of Engineering Hunedoara,
5, Revolutiei, 331128, Hunedoara, ROMANIA
http://acta.fih.upt.ro