Techno-Science Research Journal 8 (2020) 113-119

Content list available at ITC

Techno-Science Research Journal

Journal Homepage: www.ric.itc.edu.kh

Techno-Science Research Journal

Modeling, Control and Simulation on 3DOF Robot Manipulator

Leangchheng Ly*, Sarot Srang, Daro Van

Department of Industrial and Mechanical Engineering, Institute of Technology of Cambodia, Russian Federation Blvd., P.O. Box 86, Phnom Penh, Cambodia

Received: 29 June 2020; Accepted: 08 November 2020; Available online: xxxx

Abstract: *This paper describes the mathematical model, control and simulation of 3DoF (Degrees of Freedom) robot manipulator moving in a three-dimensions Cartesian coordinate frame. The four Denavit-Hartenberg (D-H) parameters are used to determine the homogenous transformations matrices of the links of the robot and express the robot's forward kinematics. Both link position and orientation are embedded in the homogeneous transformation matrix. The geometric solution approach based on the decomposing of the spatial geometry of the robot into several plane geometry is applied to determine the inverse kinematic problem of the robot. Dynamic equations of the robot are important studies to provide an understanding of the behavior of robot moving in a three-dimensional coordinate frame. The Newton-Euler formulation is utilized to recursively derive the dynamic equation of each link of the robotic manipulator, and it is used to analyze the robot by treating the system into separate parts. The forward recursion is used to derive the generalized forces working from the end link to the base of the manipulator, and backward recursion of the velocities and accelerations working from the base of the manipulator to the end link. The end result is to determine the mathematical model of the robot in an explicit form and use that explicit equation to simulate the whole system. The mathematical models of the kinematic and dynamic equations of the robot are derived, and the validity of the model is proved by numerical simulation in conjuction with control systems.*

Keywords: Kinematic; Dynamic; Simulation

1. INTRODUCTION

The manipulator is a reprogrammable, with multifunctional, and specialized devices desinged to manipulate materials without direct physical contact by the operator (Høifødt, 2011). Robots are programmed through variable motions for the performance of a variety of tasks. Today robot has more complicated applications and requires much more motion capability and sensing (Jazar, 2010). The motion of the robot can be interpreted in the form of mathematical equations such as kinematics and dynamics. The kinematic and dynamic modelings are crucial for the simulation of robot motions, and design of control algorithms (Sciavicco & Siciliano, 2012). Denavit-Hartenberg (DH) convention has been used most often in the kinematic models of the robot manipulators (Kucuk & Bingul, 2006) such as robot frame assign and robot forward kinematics determined (Fateh, 2009). The algebraic and geometric solution approaches are the two solution approaches to determine the inverse kinematics problems of the robot. The algebraic solution approach is determined by using the forward kinematics equation and pre-multiplied on the left by the link transformation inverse. The algebraic approach used to be applied to robot manipulators with many degrees of freedom or working in a three-dimensional workspace. The geometry solution approach is based on the decomposing of the spatial geometry of the robot into several plane geometry problems. However, it is applied to the simple robot structure. Lagrange-Euler and Newton-Euler formulations are well-known approaches for dynamic analysis of robot manipulators (Lynch & Park, 2017). In the standard Lagrange-Euler formulation, the robotic system is analyzed based on its kinetic and potential energy. The Newton-Euler formulation is different from the Lagrange-Euler formulation since each link of the manipulator is considered separately while deriving dynamic equations. This method has two adavantages. The recursive property makes it easy for implementation, and it induces the constraint equation for calculation of reaction forces at each join. Therefore, we use Newton-Euler formation for modeling the system.

* Corresponding author: Leangchheng Ly
E-mail: leangchheng.ly@gsc.itc.edu.kh;
Tel: +855-96-598-786-4

This paper focuses on kinematics, dynamics modeling, control and simulation model of 3DoF robot manipulator moving in three-dimensional space, and is organized as follows. In Section 2., the forward and inverse kinematic models are described. The models based on forward and inverse dynamics are given in section 3. The control and simulation of the robot are presented in section 4. The conclusion is presented in section 5.

2. KINEMATICS MODEL

The present section focuses on the development of kinematics modeling of 3dof robotic manipulator moving in the 3D Cartesian coordinate frame shown in Fig. 1. The kinematic model is to derive equations describing the motion of the robot without considering forces and torques causing the motion. Forward and inverse kinematics are the two complementary tasks in the kinematics model which are used to determine the kinematics problems of the robot.

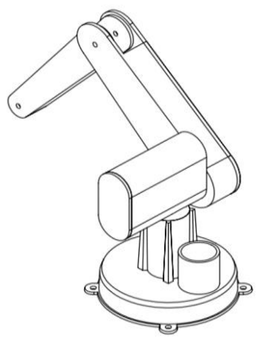

Fig. 1. 3-RRR robot manipulator in 3D view

2.1 Forward kinematics

In forward kinematics, we wants to find the position and orientation of the end-effector in the function of the given joints coordinates. The manipulator here is considered to have masses and lengths as $\{m_1, m_2, m_3\}$ and $\{L_1, L_2, L_3\}$ for link 1 to link 3, respectively. The rotation angle of the robot from the base to link 3 is given as θ_1, θ_2, and θ_3. The joints coordinates of the robot assigned by following the Denavit-Hartenberg (D-H) convention as shown in Fig. 2., where the z-axis is always in the direction of rotation of the joint.

Table 1. D-H parameters of 3dof serial robot manipulator follow with the coordinate frame in Fig. 2

Link(*i*)	a_i	α_i	d_i	θ_i
1	0	90^0	L_1	θ_1
2	L_2	0	0	θ_2
3	L_3	0	0	θ_3

where:
a_i = link length (displacement along x_i)
α_i = link twist (rotation of frame i-1 around x_i to get z_{i-1} match z_i)
d_i = link offset (displacement along z_{i-1})
θ_i = joint angle (rotation of frame i-1 around z_{i-1} to get x_{i-1} match x_i).

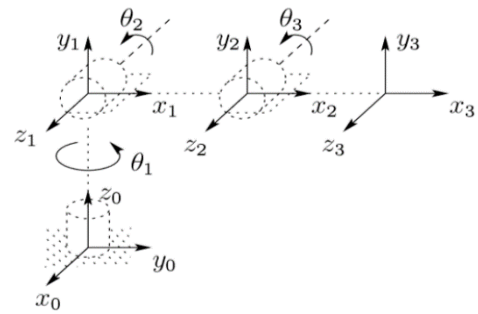

Fig. 2. The general coordinate set by the DH method (Kucuk & Bingul, 2006)

From Fig. 2., we can determine the four D-H parameters directly such as link length (a_i), link twist (α_i), link offset (d_i), and joint angle (θ_i) which are obtianed in the Table 1.

The formulation for determining the transformation matrix of link *i* of the robot is illustrated in (Eq. 1):

$$H_i^{i-1} = Rot_z(\theta_i)Tran_z(d_i)Tran_x(a_i)Rot_x(\alpha_i), \quad \text{(Eq. 1)}$$

where:
Rot_z and Rot_x present the rotation, $Tran_z$ and $Tran_z$denote the translation. The (Eq. 1) can be written in expanded form as in (Eq. 2) below:

$$H_i^{i-1} = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(Eq. 2)}$$

where:
$s\theta_i$ and $c\theta_i$ are the short hands of $\sin(\theta_i)$ and $\cos(\theta_i)$ respectively, and similarly for $s\alpha_i$ and $c\alpha_i$. The forward kinematics of the end-effector with respect to the base frame is determined by multiplying all of the H_i^{i-1} matrices:

$$H_{end-effecotr}^{base} = H_1^0 H_2^1 \dots H_n^{n-1} \quad \text{(Eq. 3)}$$

Alternatively H_n^0 can be written as:

$$H_n^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(Eq. 4)}$$

where:

r_{jk} represents the rotational elements of rotation matrix (j and k = 1, 2, 3).

p_x, p_y and p_z denote the elements of the position vector.

From Table 1., the values θ_i, α_i, d_i and a_i are used to substitute in the (Eq. 2), and the homogeneous of the link 1 to link 3 of the robot are computed respectively as in the (Eq. 5), (Eq. 6), and (Eq. 7) below:

$$H_1^0 = \begin{bmatrix} c\theta_1 & 0 & s\theta_1 & 0 \\ s\theta_1 & 0 & -c\theta_1 & 0 \\ 0 & 1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(Eq. 5)}$$

$$H_2^1 = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & L_2 c\theta_2 \\ s\theta_2 & c\theta_2 & 0 & L_2 s\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(Eq. 6)}$$

$$H_3^2 = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & L_3 c\theta_3 \\ s\theta_3 & c\theta_3 & 0 & L_3 s\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(Eq. 7)}$$

For the three DOF manipulator, the position and orientation of the end-effector with respect to the base frame are derived as:

$$H_3^0 = \begin{bmatrix} c_1 c_{23} & -c_1 s_{23} & s_1 & -L_2 c_1 c_2 - L_3 c_1 c_{23} \\ s_1 c_{23} & -s_1 s_{23} & -c_1 & L_2 s_1 c_2 + L_3 s_1 c_{23} \\ s_{23} & c_{23} & 0 & L_1 + L_2 s_2 + L_3 s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(Eq. 8)}$$

where :

s_1 and c_1 are the short hands of $\sin(\theta_1)$ and $\cos(\theta_1)$ respectively, and s_{23} and c_{23} are the short hands of $\sin(\theta_2 + \theta_3)$ and $\cos(\theta_2 + \theta_3)$. In the (Eq. 8), the orientation matrix and position vector of the end-effector are,

$$R = \begin{bmatrix} c_1 c_{23} & -c_1 s_{23} & s_1 \\ s_1 c_{23} & -s_1 s_{23} & -c_1 \\ s_{23} & c_{23} & 0 \end{bmatrix} \quad \text{(Eq. 9)}$$

$$P = \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} = \begin{pmatrix} -L_2 c_1 c_2 - L_3 c_1 c_{23} \\ L_2 s_1 c_2 + L_3 s_1 c_{23} \\ L_1 + L_2 s_2 + L_3 s_{23} \end{pmatrix} \quad \text{(Eq. 10)}$$

2.2 Inverse kinematics

In inverse kinematics, the joint variables are determined in function of the given position and orientation of the end-effector. The geometric solution approach which is based on the decomposing of the spatial geometry of the robot is used to determine the inverse kinematics of the manipulator here. The given position and orientation of the end-effector point P is shown in Fig. 3.

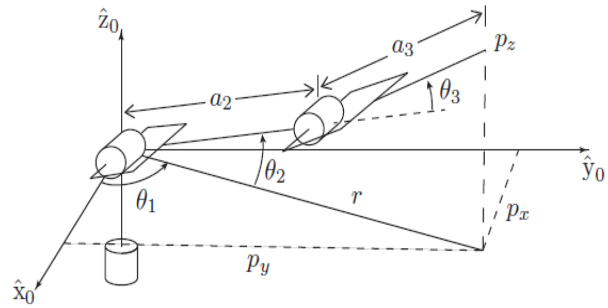

Fig. 3. Inverse position of the robot (Lynch & Park, 2017)

The variable of joint 1 of the robot can be directly defined as:

$$\theta_1 = arctan2(p_y, p_x) \quad \text{(Eq. 11)}$$

Let $h = p_z - L_1$, and $d^2 = p_x^2 + h^2$. Applying the law of cosines that applies in Fig. 4., then the variable of joint 3 is:

$$d^2 = L_2^2 + L_3^2 - 2L_2 L_3 \cos(\pi - \theta_3) \quad \text{(Eq. 12)}$$

$$\theta_3 = \arccos\left(\frac{d^2 - L_2^2 - L_3^2}{2L_2 L_3}\right) \quad \text{(Eq. 13)}$$

Similarly for the variable of joint two:

$$\beta = arccos\left(\frac{d^2 + L_2^2 - L_3^2}{2dL_2}\right) \quad \text{(Eq. 14)}$$

$$\gamma = \arctan 2(h, p_x) \quad \text{(Eq. 15)}$$

$$\theta_2 = \gamma - \beta \quad \text{(Eq. 16)}$$

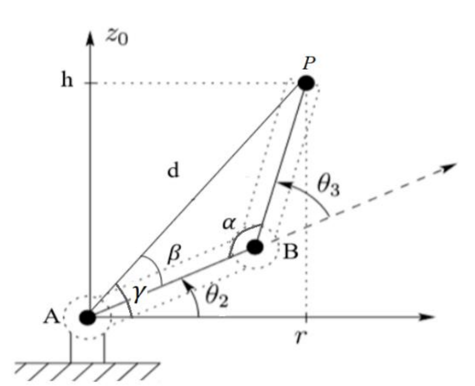

Fig. 4. Plane geometry for determining the variable of joints 2 and 3

3. DYNAMICS MODEL

Dynamic modeling means deriving equations that explicitly describes the relationship between the action forces and acceleration generated by the robot. The Newton-Euler formulation is the recursion method which is used to derive the dynamical problem of the system. The whole Newton-Euler formulation (Newton law of motion, Newton law of mechanisms) has presented in (Taylor & others, 2005).
Newton law of mechanisms:

$$\Sigma f = m_i a_i, \quad \text{(Eq. 17)}$$

$$f_i - R_{i+1}^{i} f_{i+1} + m_i g_i = m_i a_{c,i}, \quad \text{(Eq. 18)}$$

$$\Sigma \tau = \omega_i \times I_i \omega_i + \dot{\omega}_i. \quad \text{(Eq. 19)}$$

$$\tau_i - R_{i+1}^{i} \tau_{i+1} + f_i \times r_{i-1,ci} - (R_{i+1}^{i} f_i) \times r_{i,ci} = \omega_i \times I_i \omega_i + \dot{\omega}_i \quad \text{(Eq. 20)}$$

where:
m_i is the mass of link *i*, f, and τ are the force and torque acting on the link, I and ω are the moment of inertia and angular velocity of the link of the robot.
The ω of each link of the robot with respect to the base is determined by following the formula in the (Eq. 21):

$$\omega_i = \left(R_i^{i-1}\right)^T \omega_{i-1} + b_i \dot{q}_i \quad \text{(Eq. 21)}$$

where:

$$b_i = (R_i^0)^T R_{i-1}^0 z_0 \quad \text{(Eq. 22)}$$

is the rotation of joint *i* expressed in frame *i*, q_i is the generalized coordinate of the joint, R_i^{i-1} is the rotation matrix of frame *i-1* to frame *i*, and z_0 is the axis of actuation of frame zero (base).

The angular acceleration (α) of each link of the robot with respect to the base is determined by following the formula in the (Eq. 23):

$$\alpha_i = (R_i^0)^T \dot{\omega}_i \quad \text{(Eq. 23)}$$

where:

$$\dot{\omega}_i = \dot{\omega}_{i-1} + b_i \ddot{q}_i + \omega_i \times b_i \dot{q}_i \quad \text{(Eq. 24)}$$

is the time derivative of the ω_i.

The acceleration of the center of mass ($a_{c,i}$) and the end of link *i* ($a_{e,i}$), expressed in frame *i*, are defined as in (Eq. 25) and (Eq. 26), respectively:

$$a_{c,i} = \left(R_i^{i-1}\right)^T a_{e,i-1} + \dot{\omega}_i \times r_{i-1,ci} + \omega_i \times \left(\omega_i \times r_{i-1,ci}\right), \quad \text{(Eq. 25)}$$

$$a_{e,i} = \left(R_i^{i-1}\right)^T a_{e,i-1} + \dot{\omega}_i \times r_{i-1,i} + \omega_i \times \left(\omega_i \times r_{i-1,i}\right). \quad \text{(Eq. 26)}$$

where:
$r_{i-1,ci}$ is the vector from the origin of frame *i* - 1 to the center of mass of link *i*, $r_{i,ci}$ is the vector from the origin of frame *i* to the center of mass of link *i*, and $r_{i-1,i}$ is the vector from the origin of frame *i* - 1 to the origin of frame *i*.

From Fig. 2., the axis of actuation $z_0 = [0 \quad 0 \quad 1]^T$, the vector $g_0 = [0 \quad 0 \quad -g]^T$, and the rotation matrix R_i^{i-1} is already computed in section 3.

3.1 Forward recursion

The linear and angular motion of the links is calculated by applying the forward recursion Newton formulation in (Eq. 21), (Eq. 23), (Eq. 25) and (Eq. 26) respectively. The forward recursion is starting with link 1 and ending with link 3.The initial condition is $\omega_0 = \alpha_0 = a_{c,0} = a_{e,0} = 0$.

Link 1
The angular velocity and angular acceleration of link 1 are:

$$\omega_1 = b_1 \dot{\theta}_1 \quad \text{(Eq. 27)}$$

$$\alpha_1 = b_1 \ddot{q}_1 + \omega_1 \times b_1 \dot{q}_1 \quad \text{(Eq. 28)}$$

The acceleration of the center of mass ($a_{c,1}$) and the end of link 1 ($a_{e,1}$) are:

$$a_{c,1} = \dot{\omega}_1 \times r_{0,c1} + \omega_1 \times \left(\omega_1 \times r_{0,c1}\right) \quad \text{(Eq. 29)}$$

$$a_{e,1} = \dot{\omega}_1 \times r_{0,1} + \omega_1 \times \left(\omega_1 \times r_{0,1}\right) \quad \text{(Eq. 30)}$$

Link 2
The angular velocity and angular acceleration of link 2 are:

$$\omega_2 = (R_2^1)^T \omega_1 + b_2 \dot{\theta}_2 \quad \text{(Eq. 31)}$$

$$\alpha_2 = (R_2^1)^T \alpha_1 + b_2 \ddot{q}_2 + \omega_2 \times b_2 \dot{q}_2 \quad \text{(Eq. 32)}$$

The acceleration of the center of mass ($a_{c,2}$) and the end of link 2 ($a_{e,2}$) are:

$$a_{c,2} = (R_2^1)^T a_{e,1} + \dot{\omega}_2 \times r_{1,c2} + \omega_2 \times \left(\omega_2 \times r_{1,c2}\right) \quad \text{(Eq. 33)}$$

$$a_{e,1} = (R_2^1)^T a_{e,1} + \dot{\omega}_2 \times r_{1,2} + \omega_2 \times (\omega_2 \times r_{1,2}) \quad \text{(Eq. 34)}$$

Link 3

The angular velocity and angular acceleration of link 3 are:

$$\omega_3 = (R_3^2)^T \omega_2 + b_3 \dot{\theta}_3 \quad \text{(Eq. 35)}$$

$$\alpha_3 = (R_3^2)^T \alpha_2 + b_3 \ddot{q}_3 + \omega_3 \times b_3 \dot{q}_3 \quad \text{(Eq. 36)}$$

The acceleration of the center of mass ($a_{c,3}$) and the end of link 3 ($a_{e,3}$) are:

$$a_{c,3} = (R_3^2)^T a_{e,2} + \dot{\omega}_3 \times r_{2,c3} + \omega_3 \times (\omega_3 \times r_{2,c3}), \quad \text{(Eq. 37)}$$

$$a_{e,3} = (R_3^2)^T a_{e,2} + \dot{\omega}_3 \times r_{2,3} + \omega_3 \times (\omega_3 \times r_{2,3}) \quad \text{(Eq. 38)}$$

3.2 Backward recursion

The forces and joint torques acting on the links are calculated by applying the backward recursion Newton formulation in (Eq. 18) and (Eq. 20) respectively. The backward recursion is starting with link 3 and ending with link 1. The terminal condition is $f_4 = \tau_4 = 0$.

Link 3

The gravity vector of link 3 is $g_3 = (R_3^0)^T g_0$. The force and joint torque exerting on the link are:

$$f_3 = m_3 a_{c,3} - m_3 g_3, \quad \text{(Eq. 39)}$$

$$\tau_3 = -f_3 \times r_{2,c3} + \omega_3 \times (I_3 \omega_3) + I_3 \dot{\alpha}_3 \quad \text{(Eq. 40)}$$

Link 2

The gravity vector of link 2 is $g_2 = (R_2^0)^T g_0$. The force and joint torque exerting on the link are:

$$f_2 = R_3^2 f_3 + m_2 a_{c,2} - m_2 g_2 \quad \text{(Eq. 41)}$$

$$\tau_2 = R_3^2 \tau_3 - f_2 \times r_{1,c2} + R_3^2 f_3 \times r_{2,c2} + \omega_2 \times (I_2 \omega_2) + I_2 \dot{\alpha}_2 \quad \text{(Eq. 42)}$$

Link 1

The gravity vector of link 1 is $g_1 = (R_1^0)^T g_0$. The force and joint torque exerting on the link are:

$$f_1 = R_2^1 f_2 + m_1 a_{c,1} - m_1 g_1 \quad \text{(Eq. 43)}$$

$$\tau_1 = R_2^1 \tau_2 - f_1 \times r_{0,c1} + R_2^1 f_2 \times r_{1,c1} + \omega_1 \times (I_1 \omega_1) + I_1 \dot{\alpha}_1 \quad \text{(Eq. 44)}$$

By substituting the angular velocity, angular acceleration and link acceleration that are computed in the forward recursion in the backward recursion equation, the dynamics model of the robot is determined. To be more convenient for simulation, the dynamics model of the robot is written in an compact form as:

$$M(q)\ddot{q} + h(q, \dot{q}) = \mathrm{T} \quad \text{(Eq. 45)}$$

where:
$M(q)$ is the inertia matrix, q is the generalized coordinate of the joint, T is the torque vector, and $h(q, \dot{q}) = C(q, \dot{q}) + G(q)$ is the Coriolis and Centrifugal, and the gravity term respectively. The element of the matrices M, C, and G are found respectively below:

$$M_{11} = \tfrac{1}{2}\big(I_{2x} + I_{3x} + 2I_{1y} + I_{2y} + I_{3y} - I_{2x}\cos(2\theta_2) + I_{2y}\cos(2\theta_2) - I_{3x}\cos(2\theta_2 + 2\theta_3) + I_{3y}\cos(2\theta_2 + 2\theta_3)\big) + \tfrac{1}{4}(L_3^2 m_3 \cos(\theta_2 + \theta_3) + L_2^2 m_3 \cos(\theta_2) + L_2^2 m_2 \cos(\theta_2) + 2L_2 L_3 m_3 \cos(\theta_2 + \theta_3) + 2L_2 L_3 m_3 \cos(\theta_2)),$$

$$M_{12} = M_{13} = M_{21} = 0,$$

$$M_{22} = I_{2z} + I_{3z} + L_2^2 m_3 + \tfrac{1}{4}(L_2^2 m_2 + L_3^2 m_3) + L_2 L_3 m_3 \cos(\theta_3),$$

$$M_{23} = M_{32} = I_{3z} + \tfrac{1}{4} L_3^2 m_3 + \tfrac{1}{2} L_2 L_3 m_3 \cos(\theta_3),$$

$$M_{31} = 0,$$

$$M_{33} = I_{3z} + \tfrac{1}{4} L_3^2 m_3.$$

$$C_1 = \dot{\theta}_i \dot{\theta}_2 \Big(I_{3x}\sin(2\theta_2 + 2\theta_3) - I_{3y}\sin(2\theta_2 + 2\theta_3) + I_{2x}\sin(2\theta_2) - I_{2y}\sin(2\theta_2) + \tfrac{1}{8} L_3^2 m_3 \sin(\theta_3) - \tfrac{1}{8} L_3^2 m_3 \sin(2\theta_2 + \theta_3) - \tfrac{1}{4} L_2^2 m_2 \sin(2\theta_2) - L_2^2 m_3 \sin(2\theta_2) - \tfrac{1}{8} L_3^2 m_3 \sin(2\theta_2 + 2\theta_3) + \tfrac{1}{4} L_2 L_3 m_3 \sin(\theta_3) - \tfrac{3}{4} L_2 L_3 m_3 \sin(2\theta_2 + \theta_3) - \tfrac{1}{4} L_2 L_3 m_3 \sin(2\theta_2)\Big) + \dot{\theta}_1 \dot{\theta}_3 \Big(I_{3x}\sin(2\theta_2 + 2\theta_3) - I_{3y}\sin(2\theta_2 + 2\theta_3) - \tfrac{1}{4} L_3^2 m_3 \sin(2\theta_2 + 2\theta_3) - \tfrac{1}{2} L_2 L_3 m_3 \sin(\theta_3) - \tfrac{1}{2} L_2 L_3 m_3 \sin(2\theta_2 + \theta_3)\Big),$$

$$C_2 = \dot{\theta}_1^2 \Big(\tfrac{1}{2} I_{3y}\sin(2\theta_2 + 2\theta_3) - \tfrac{1}{2} I_{3x}\sin(2\theta_2 + 2\theta_3) - \tfrac{1}{2} I_{2x}\sin(2\theta_2) + \tfrac{1}{2} I_{2y}\sin(2\theta_2) + \tfrac{1}{8} L_2^2 m_2 \sin(2\theta_2) + \tfrac{1}{2} L_2^2 m_3 \sin(2\theta_2) + \tfrac{1}{8} L_3^2 m_3 \sin(2\theta_2 + 2\theta_3) + \tfrac{1}{2} L_2 L_3 m_3 \sin(2\theta_2 + \theta_3)\Big) - \tfrac{1}{2} L_2 L_3 m_3 \sin(\theta_3)\, \dot{\theta}_3^2 - L_2 L_3 m_3 \sin(\theta_3)\, \dot{\theta}_2 \dot{\theta}_3,$$

$$C_3 = \dot{\theta}_1^2 \Big(\tfrac{1}{2} I_{3y}\sin(2\theta_2 + 2\theta_3) - \tfrac{1}{2} I_{3x}\sin(2\theta_2 + 2\theta_3) + \tfrac{1}{8} L_3^2 m_3 \sin(2\theta_2 + 2\theta_3) + \tfrac{1}{4} L_2 L_3 m_3 \sin(\theta_3) + \tfrac{1}{2} L_2 L_3 m_3 \sin(2\theta_2 + \theta_3)\Big) + \tfrac{1}{2} L_2 L_3 m_3 \sin(\theta_3)\, \dot{\theta}_2^2.$$

$$G_1 = 0;$$

$$G_2 = \frac{1}{2} g(L_2 m_2 \cos(\theta_2) + 2L_2 m_3 \cos(\theta_2) + L_3 m_3 \cos(\theta_2 + \theta_3)),$$

$$G_3 = \frac{1}{2} g L_3 m_3 \cos(\theta_2 + \theta_3)$$

4. CONTROL AND SIMULATION

The Simulink model in Fig. 5. is designed to verify the dynamics model which is derived in section 3. This model is simulated as the ideal case of the system with no feedback from the output. The output of this simulation is the joints positions and joints velocities correspondence to the zero torques applied. The initial condition is $q_{\_int} = [0\ 0\ 0\ 0\ 0\ 0]^T$.

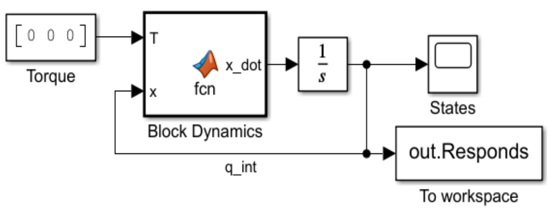

Fig. 5. Open loop Simulink model

The response of joints 2 and 3 in Fig. 6., both positions and velocities are very oscillated around its equilibrium point due to the excitation of gravity on link 2 and 3. But the response of joint 1 is zero because gravity does not affect link 1 of the robot.

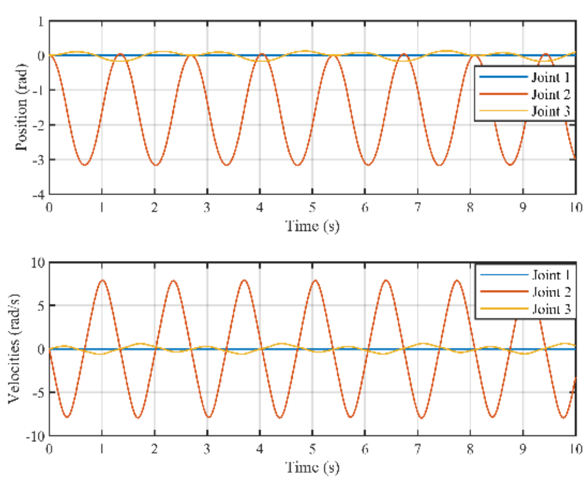

Fig. 6. The opened loop response with zero torque was applied

The Simulink model in Fig. 7., built in a closed-loop system with the velocity feedforward, position feedback control architecture, and PID controller. The system is simulated by assuming that the dynamics of the motors and joints friction are known. The output results are joints positions and joints velocities. The joints velocities are used to feedforward to the system to regulate the response track to the desire. The simulation is studied on two conditions, simulated with the constant input 1 and cosine wave. PI control is used in the simulation procedure.

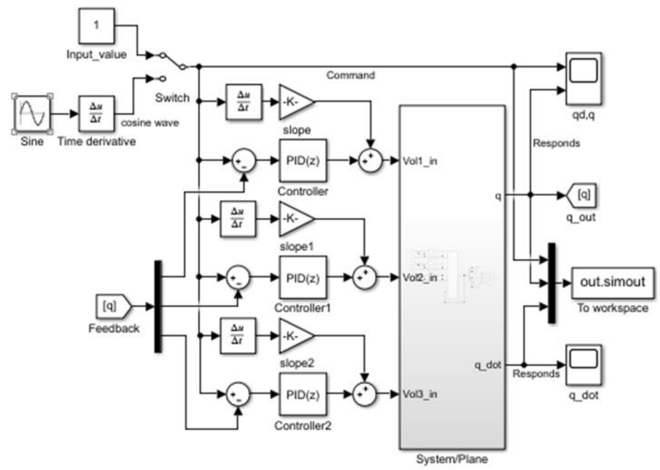

Fig. 7. Closed-loop Simulink model with PID controller

The simulation results of the system with a constant 1 and zero are used as the desire joints positions and joints velocities shown in Fig. 8. The responses of the system track the desires.

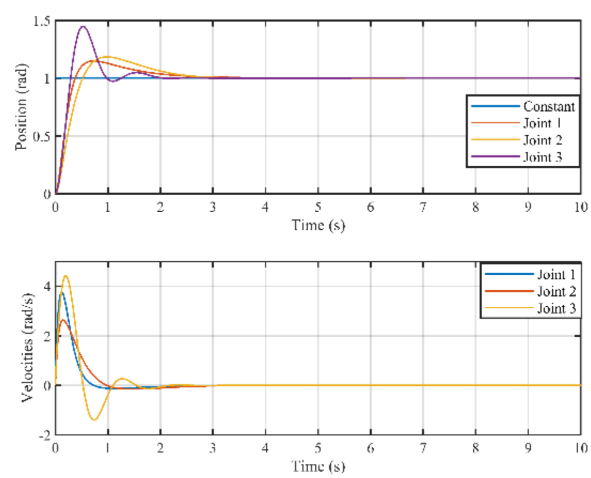

Fig. 8. The response of the closed-loop system with a constant 1 and its derivative is used as an input

The cosine wave and its derivative are used as the desire joints positions and joints velocities. The simulation result is shown in Fig. 9. The responses of the system asymptotically track the desires.

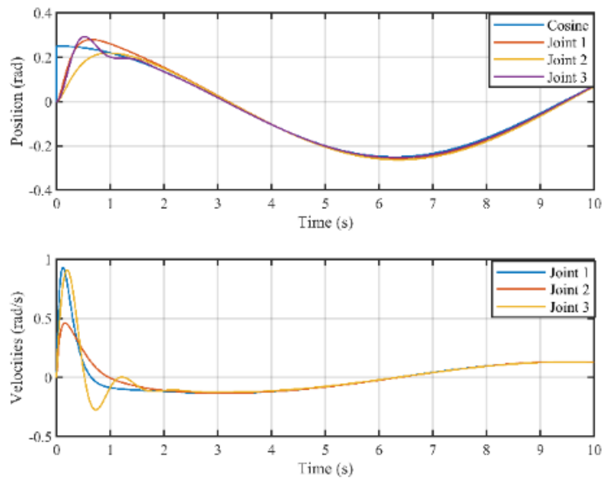

Fig. 9. The response of the closed-loop system with $\mathbf{cos}(\boldsymbol{t})$ and its derivative is used as an input

5. CONCLUSIONS

In this work, the mathematical modeling of the system of 3DOF robot manipulator which moves in three-dimensional working space is completely derived by using Newton-Euler formulation. This provides the basic concept for mechanical design, controller design, and simulation. The dynamic model allows us to compute the joints variable in function of forces and applied or compute the forces/torques of the robot in the function of joints variables. The simulation model allows us to verify the mathematics model and to design the controller for the robot. The results of the simulation show that the system with PI asymptotically tracks to the desires in both cases of constant and a cosine wave. The further research will be focusing on designing the controller by using velocity and acceleration feedforward, position feedback control architecture, and build the prototype of the 3 DOF robot manipulator. Moreover, the prototyping will be used as the concept to design and build the legged robot for which each leg uses the 3DOF robot manipulator. Furthermore, design and build a robot manipulator with more degrees of freedom which can work with a variety of tasks in widely workspaces.

ACKNOWLEDGMENTS

I would like to express my deep gratitude to the ARES-CCD scholarship program for full support on tuition fees and monthly allowance. I also would like to thank all Dynamics and Control Laboratory members that always share some of all their experiences, and knowledge that relate to my project.

REFERENCES

Fateh, M. M. (2009). Dynamic Modeling of Robot Manipulators in DH Frames. *World Applied Sciences Journal*, *6(1)*, 39-44.

Høifødt, H. (2011). Dynamic modeling and simulation of robot manipulators. *Master of Science in Engineering Cybernetics, Norwegian University of Science and Technology*.

Jazar, R. (2010). *Theory of applied robotics: kinematics, dynamics, and control.* Springer Science \& Business Media.

Kucuk, S., & Bingul, Z. (2006). *Robot kinematics: Forward and inverse kinematics.* INTECH Open Access Publisher.

Lynch, K., & Park, F. (2017). *Modern Robotics.* Cambridge University Press.

Sciavicco, L., & Siciliano, B. (2012). *Modelling and control of robot manipulators.* Springer Science \& Business Media.

Taylor, J. R., & others. (2005). *Classical mechanics.* University Science Books.