



# Dynamic Modeling and Control of a Flexible Link Manipulators with Translational and Rotational Joints

Duong Xuan Bien<sup>1,\*</sup>, Chu Anh My<sup>1</sup>, Phan Bui Khoi<sup>2</sup>

<sup>1</sup>*Military Technical Academy, No. 236, Hoang Quoc Viet, Cau Giay, Hanoi, Vietnam*

<sup>2</sup>*Hanoi University of Science and Technology, No. 1, Dai Co Viet, Hai Ba Trung, Hanoi, Vietnam*

Received 08 December 2017

Revised 04 January 2018; Accepted 04 January 2018

**Abstract:** Flexible link manipulators are widely used in many areas such the space technology, medical, defense and automation industries. They are more realistic than their rigid counterparts in many practical conditions. Most of the investigations have been confined to manipulators with only rotational joint. Combining such systems with translational joints enables these manipulators more flexibility and more applications. In this paper, a nonlinear dynamic modeling and control of flexible link manipulator with rigid translational and rotational joints is presented. This model TR (Translational-Rotational) is developed based on single flexible link manipulator with only rotational joint. Finite element method and Lagrange approach are used to model and build equations of the motion. PID controller is designed with parameters ( $K_p$ ,  $K_i$ ,  $K_d$ ) which are optimized by using Particle Swarm Optimization algorithm (PSO). Errors of joints variables and elastic displacements at the end-effector point are reduced to warrant initial request. The results of this study play an important role in modeling generalized planar flexible two-link robot and in selecting the suitable structure robot with the same request.

**Keywords:** Flexible link, translational joint, elastic displacements, control, PSO.

## 1. Introduction

Flexible link manipulators with translational and rotational joint have received more attention recently because of many advantages and applications. The considering translational joint and elastic displacements effects on robot motion become complicated because of highly nonlinear characteristics.

Few authors have studied the manipulator with only translational joint. Wang and Duo Wei [1] presented a single flexible robot arm with translational joint. Dynamic model analysis is based on a Galerkin approximation with time dependent basis functions. They also proposed a feedback control

\*Corresponding author. Tel.: 84- 1667193567.

Email: [xuanbien82@yahoo.com](mailto:xuanbien82@yahoo.com)

<https://doi.org/10.25073/2588-1124/vnumap.4240>

law in [2]. Kwon and Book [3] present a single link robot which is described and modeled by using assumed modes method (AMM). Other authors have focused on the flexible manipulator with a link slides through a translational joint with a simultaneous rotational motion (R-T robot). Pan et al [4] presented a model R-T with FEM method. The result is differential algebraic equations which are solved by using Newmark method. Yuh and Young [5] proposed the partial differential equations with R-T system by using AMM. Al-Bedoor and Khulief [6] presented a general dynamic model for R-T robot based on FEM and Lagrange approach. They defined a concept which is translational element. The stiffness of translational element is changed. The translational joint variable is distance from origin coordinate system to translational element. The number of element is small because it is hard challenge to build and solve differential equations. Khadem [7] studied a three-dimensional flexible n-degree of freedom manipulator having both revolute and translational joint. A novel approach is presented using the perturbation method. The dynamic equations are derived using the Jourdain's principle and the Gibbs-Appell notation. Korayem [8] also presented a systematic algorithm capable of deriving equations of motion of N-flexible link manipulators with revolute-translational joints by using recursive Gibbs-Appell formulation and AMM.

In addition, the order of the translational joints in the kinematic chain has not been considered in the reviewed researches. Almost related works demonstrate their method through the rotational – translational model (R -T model). This is just a specific case of the general kinematic chain of the flexible manipulator.

There are many researchers who focused on intelligent control system development to end-effectors control as Fuzzy Logic [10], Neural Network [11], PSO [12], Back-stepping [13] and Genetic Algorithm [14]. PSO was formulated by Edward and Kennedy in 1995. PSO algorithm is optimization technique by social behavior of bird flocking [15]. This technique is similar to the continuous genetic algorithm (GA) in that it begins with a random population matrix. Unlike the GA, PSO has no evolution operators such as crossover and mutation. PSO Optimum solution is found by sharing information in the search space. This is a population based search algorithm which is initialized with the population of random solutions, called particles and the population is known as swarm [15]. The main strength of PSO is that it is easy to implement and fast convergent. PSO has become robust and widely applied in continuous and discrete optimization for engineering applications.

However, most of the investigations on intelligent control of the flexible robot manipulator focus on the robot structure constructed with all rotational joints.

In this work, dynamic model of flexible link manipulator combining translational and rotational joints is presented. This model (T-R) is difference R-T model. The first link is assumed rigidly which is attached rigid translational. The second link is flexibility with rigid rotational joint. T-R model has not been mentioned yet before. The dynamic model is described in fig. 1. The PID control system is designed to warrant following reference point and desire path in joint space based on errors of joint variables and value of elastic displacement at the end-effector point. Parameters of PID control are optimized by using PSO algorithm. Fitness function is the linear quadratic regulator (LQR) function.

## 2. Dynamic modeling and equations of motion

### 2.1. Dynamic modeling

The model of two link flexible robot which motions on horizontal plane with translational joint for first rigid link and rotational joint for second flexible link is shown as Fig 1.

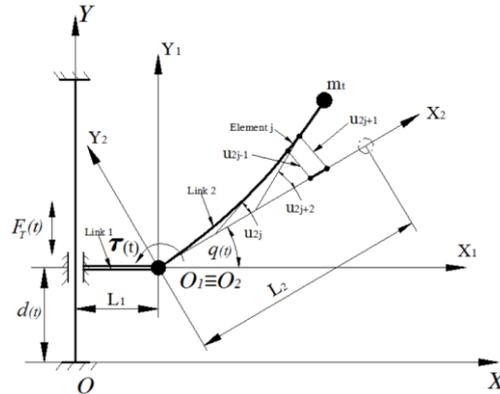


Fig. 1. Flexible links robot with translational and rotational joints.

The coordinate system  $XOY$  is the fixed frame. Coordinate system  $X_1O_1Y_1$  is attached to end point of link 1. Coordinate system  $X_2O_2Y_2$  is attached to first point of link 2. The translational joint variable  $d(t)$  is driven by  $F_T(t)$  force. The rotational joint variable  $q(t)$  is driven by  $\tau(t)$  torque. Both joints are assumed rigid. Link 1 with length  $L_1$  is assumed rigid and link 2 with length  $L_2$  is assumed flexibility. Link 2 is divided into  $n$  elements. The elements are assumed interconnected at certain points, known as nodes. Each element has two nodes. Each node of element  $j$  has 2 elastic displacement variables which are the flexural  $(u_{2j-1}, u_{2j+1})$  and the slope displacements  $(u_{2j}, u_{2j+2})$ . Symbol  $m_i$  is the mass of payload on the end-effector point. The coordinate  $r_{o1}$  of end point of link 1 on  $XOY$  is computed as

$$r_{o1} = [L_1 \quad d(t)]^T \tag{1}$$

The coordinate  $r_{2j}$  of element  $j$  on  $X_2O_2Y_2$  can be given as

$$r_{2j} = [(j-1)l_e + x_j \quad w_j(x_j, t)]^T ; \tag{2}$$

$$(0 \leq x_j \leq l_e)$$

Where, length of each element is  $l_e = L_2/n$  and  $w_j(x_j, t)$  is the total elastic displacement of element  $j$  which is defined by [9]

$$w_j(x_j, t) = N_j(x_j)Q_j(t) \tag{3}$$

Vector of shape function  $N_j(x_j)$  is defined as

$$N_j(x_j) = [\phi_1 \quad \phi_2 \quad \phi_3 \quad \phi_4] \tag{4}$$

Mode shape function  $\phi_i(x_j); (i = 1 \dots 4)$  can be presented in [9]. The elastic displacement  $Q_j(t)$  of element  $j$  is given as

$$Q_j(t) = [u_{2j-1} \quad u_{2j} \quad u_{2j+1} \quad u_{2j+2}]^T \tag{5}$$

Coordinate  $r_{21j}$  of element  $j$  on  $X_1O_1Y_1$  can be written as

$$r_{21j} = T_2^j \cdot r_{2j} \tag{6}$$

Where,  $T_2^j = \begin{bmatrix} \cos q(t) & -\sin q(t) \\ \sin q(t) & \cos q(t) \end{bmatrix}$  is the transformation matrix from  $X_2O_2Y_2$  to  $X_1O_1Y_1$ . The coordinate  $r_{02j}$  of element  $j$  on  $XOY$  can be computed as

$$r_{02j} = r_1 + r_{21j} \tag{7}$$

Elastic displacement  $Q_n(t)$  of element  $n$  is given as

$$Q_n(t) = [u_{2n-1} \quad u_{2n} \quad u_{2n+1} \quad u_{2n+2}]^T \tag{8}$$

Coordinate  $r_{0E}$  of end point of flexible link 2 on  $XOY$  can be computed as

$$r_{0E} = \begin{bmatrix} L_1 + L_2 \cos q(t) - u_{2n+1} \sin q(t) \\ d(t) + L_2 \sin q(t) + u_{2n+1} \cos q(t) \end{bmatrix} \tag{9}$$

If assumed that robot with all of links are rigid, the coordinate  $r_{0E\_rigid}$  on  $XOY$  can be written as

$$r_{0E\_rigid} = \begin{bmatrix} L_1 + L_2 \cos q(t) \\ d(t) + L_2 \sin q(t) \end{bmatrix} \tag{10}$$

The kinetic energy of link 1 can be computed as

$$T_1 = \frac{1}{2} m_1 \cdot \dot{r}_{01}^T \cdot \dot{r}_{01} \tag{11}$$

Where,  $m_1$  is the mass of link 1. The kinetic energy of element  $j$  of link 2 is determined as

$$T_{2j} = \frac{1}{2} \int_0^{l_e} m_2 \left[ \frac{\partial r_{02j}}{\partial t} \right]^2 dx_j = \frac{1}{2} \dot{Q}_{jg}^T(t) M_j \dot{Q}_{jg}(t) \tag{12}$$

Where,  $m_2$  is mass per meter of link 2. The generalized elastic displacement  $Q_{jg}(t)$  of element  $j$  is given as

$$Q_{jg} = [d \quad q \quad u_{2j-1} \quad u_{2j} \quad u_{2j+1} \quad u_{2j+2}]^T \tag{13}$$

Each element of inertial mass matrix  $M_j$  can be computed as

$$M_j(s, e) = \int_0^{l_e} m_2 \left[ \frac{\partial r_{02j}}{\partial Q_{js}} \right]^T \left[ \frac{\partial r_{02j}}{\partial Q_{je}} \right] dx_j; \tag{14}$$

( $s, e = 1, 2, \dots, 6$ )

Where,  $Q_{js}$  and  $Q_{je}$  are the  $s^{th}, e^{th}$  element of  $Q_{jg}$  vector. It can be shown that  $M_j$  is of the form

$$M_j = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} \\ m_{31} & m_{32} & & & & \\ m_{41} & m_{42} & & M_{j\_base} & & \\ m_{51} & m_{52} & & & & \\ m_{61} & m_{62} & & & & \end{bmatrix} \tag{15}$$

With,

$$M_{base} = \begin{bmatrix} 13/35 m_2 l_e & 11/210 m_2 l_e^2 & 9/70 m_2 l_e & -13/420 m_2 l_e^2 \\ 11/210 m_2 l_e^2 & 1/105 m_2 l_e^3 & 13/420 m_2 l_e^2 & -1/140 m_2 l_e^3 \\ 9/70 m_2 l_e & 13/420 m_2 l_e^2 & 13/35 m_2 l_e & -11/210 m_2 l_e^2 \\ -13/420 m_2 l_e^2 & -1/140 m_2 l_e^3 & -11/210 m_2 l_e^2 & 1/105 m_2 l_e^3 \end{bmatrix} \tag{16}$$

And,

$$m_{11} = m_2 l_e; m_{13} = m_{15} = \frac{1}{2} m_2 l_e \cos q;$$

$$m_{12} = -\frac{1}{12} m_2 l_e \left[ \begin{matrix} (6u_{2j-1} + 6u_{2j+1} + l_e u_{2j} \\ -l_e u_{2j+2}) \sin q + 6l_e (1 - 2j) \cos q \end{matrix} \right];$$

$$m_{14} = m_{16} = \frac{1}{2} m_2 l_e^2 \cos q; m_{21} = m_{12};$$

$$m_{23} = \frac{1}{20} m_2 l_e^2 (10j - 7); m_{24} = \frac{1}{60} m_2 l_e^3 (5j - 3);$$

$$m_{25} = \frac{1}{20} m_2 l_e^2 (10j - 3); m_{26} = -\frac{1}{60} m_2 l_e^3 (5j - 2);$$

$$m_{22} = \frac{1}{210} m_2 l_e \left( \begin{matrix} 210l_e^2 j(j-1) + 70l_e^2 + 54u_{2j-1}u_{2j+1} \\ + l_e^2 (2u_{2j}^2 - 3u_{2j}u_{2j+2} + 2u_{2j+2}^2) \\ + 22l_e (u_{2j-1}u_{2j} - u_{2j+1}u_{2j+2}) \\ + 13l_e (u_{2j}u_{2j+1} - u_{2j-1}u_{2j+2}) \\ + 78(u_{2j-1}^2 + u_{2j+1}^2) \end{matrix} \right);$$

$$m_{31} = m_{13}; m_{32} = m_{23}; m_{41} = m_{14}; m_{42} = m_{24};$$

$$m_{51} = m_{15}; m_{52} = m_{25}; m_{61} = m_{16}; m_{62} = m_{26}$$

The total elastic kinetic energy of link 2 is yielded as

$$T_{dh} = \sum_{j=1}^n T_{2j} = \frac{1}{2} \dot{Q}^T(t) M_{dh} \dot{Q}(t) \tag{17}$$

Inertial mass matrix  $M_{dh}$  is constituted from matrices of elements follow FEM theory, respectively. Vector  $Q(t)$  represents the generalized coordinate of system and is given as

$$Q = [d \quad q \quad u_1 \quad \dots \quad u_{2n+1} \quad u_{2n+2}]^T \tag{18}$$

Kinetic energy of payload is given as

$$T_P = \frac{1}{2} m_i \cdot \dot{r}_{0E}^T \cdot \dot{r}_{0E} \tag{19}$$

Kinetic energy of system is determined as

$$T = T_I + T_{dh} + T_P = \frac{1}{2} \dot{Q}^T(t) M \dot{Q}(t) \tag{20}$$

Matrix  $M$  is mass matrix of system. The gravity effects can be ignored as the robot movement is confined to the horizontal plane. Defining  $E$  and  $I$  are Young’s modulus and inertial moment of link 2, the elastic potential energy of element  $j$  is shown as  $P_j$  with the stiffness matrix  $K_j$  and presented as [9]

$$P_j = \frac{1}{2} \int_0^{l_e} EI \left[ \frac{\partial^2 w_j(x_j, t)}{\partial x_j^2} \right]^2 dx_j = \frac{1}{2} Q_j^T \cdot K_j \cdot Q_j \tag{21}$$

With,

$$K_j = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 12EI/l_e^3 & 6EI/l_e^2 & -12EI/l_e^3 & 6EI/l_e^2 \\ 0 & 0 & 6EI/l_e^2 & 4EI/l_e & -6EI/l_e^2 & 2EI/l_e \\ 0 & 0 & -12EI/l_e^3 & -6EI/l_e^2 & 12EI/l_e^3 & -6EI/l_e^2 \\ 0 & 0 & 6EI/l_e^2 & 2EI/l_e & -6EI/l_e^2 & 4EI/l_e \end{bmatrix} \tag{22}$$

Total elastic potential energy of system is yielded as

$$P = \sum_{j=1}^n P_j = \frac{1}{2} Q^T(t) K Q(t) \tag{23}$$

Stiffness matrix  $K$  is constituted from matrices of elements follow FEM theory similar  $M$  matrix, respectively.

## 2.2. Equations of motion

Fundamentally, the method relies on the Lagrange equations with Lagrange function  $L=T-P$  are given by

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{Q}(t)} \right)^T - \left( \frac{\partial L}{\partial Q(t)} \right)^T = F(t) \quad (24)$$

Vector  $F(t)$  is the external generalized forces acting on specific generalized coordinate  $Q(t)$  and is determined as

$$F(t) = [F_T(t) \quad \tau(t) \quad 0 \quad \dots \quad 0 \quad 0]^T \quad (25)$$

Size of matrices  $M, K$  is  $(2n+4) \times (2n+4)$  and size of  $F(t)$  and  $Q(t)$  is  $(2n+4) \times 1$ . The rotational joint of link 2 is constrained so that the elastic displacements of first node of element 1 on link 2 can be zero. Thus variables  $u_1, u_2$  are zero. By enforcing these boundary conditions and FEM theory, the generalized coordinate  $Q(t)$  becomes

$$Q = [d \quad q \quad u_3 \quad \dots \quad u_{2n+1} \quad u_{2n+2}]^T \quad (26)$$

So now, size of matrices  $M, K$  is  $(2n+2) \times (2n+2)$  and size of  $F(t)$  and  $Q(t)$  is  $(2n+2) \times 1$ . When kinetic and potential energy are known, it is possible to express Lagrange equations as shown

$$M \cdot \ddot{Q} + C \cdot \dot{Q} + K \cdot Q = F(t) \quad (27)$$

Where, the Coriolis matrices  $C$  is calculated as

$$C \cdot \dot{Q} = \dot{M} \cdot \dot{Q} - \frac{1}{2} \left( \frac{\partial}{\partial Q} (\dot{Q}^T \cdot M \cdot \dot{Q}) \right) \quad (28)$$

Structural damping is ignored in this paper.

## 3. PID controller and PSO algorithm

The PID controller has been widely used in the industry but it is hard to determine the optimal or near optimal PID parameters using classical tuning methods as Ziegler Nichols. This paper presents the PSO algorithm to find the suitable parameters of the PID controller. Each particle moves about the cost surface with a velocity. The particles update their velocities and positions based on the local and global best solutions. Fig. 2 shows the movement of a single particle ( $i$ ) at the time step  $t$  in search space. At time step ( $t$ ), the position, velocity, personal best and global best are indicated as  $x_i(t), v_i(t), p_i(t)$  and  $p_g(t)$ , respectively. The velocity  $v_i(t)$  serves as a memory of the previous flight direction, can be seen as momentum. At time step ( $t+1$ ), the new position  $x_i(t+1)$  can be calculated based on three components which are inertia, cognitive and social component.

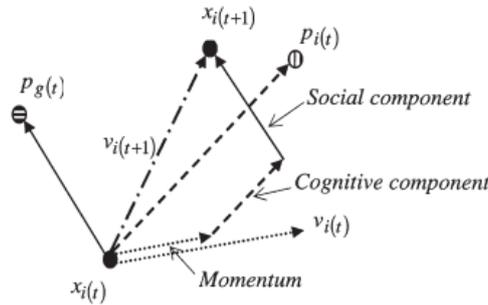


Fig. 2. The movement of a single particle

After finding the personal best and global best, particle is then accelerated toward those two best values by updating the particle position and velocity for the next iteration using the following set of equations which are given as

$$v_i(t) = kv_i(t-1) + C_1 \cdot rand \cdot (P_i - x_i(t-1)) + C_2 \cdot rand \cdot (P_g - x_i(t-1)) \tag{29}$$

And,

$$x_i(t) = x_i(t-1) + v_i(t) \tag{30}$$

Where,  $C_1$  and  $C_2$  are learning factors. Symbol *rand* is the random number between 0 and 1. Symbol  $k$  is the inertia serves as memory of the previous direction, preventing the particle from drastically changing direction. The information details of PSO algorithm can be considered as [15]. The sequences of operation in PSO are described in fig. 3 with variable *par* is the optimum solution.

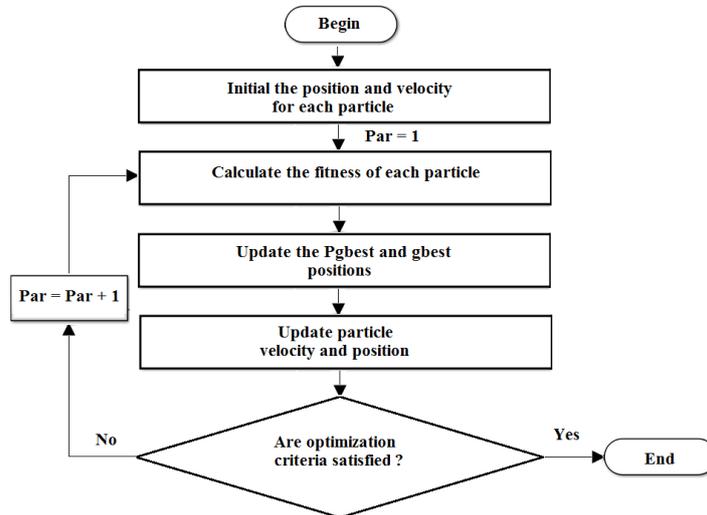


Fig. 3. Steps in PSO algorithm.

Structural controller of system is designed as in fig. 4.

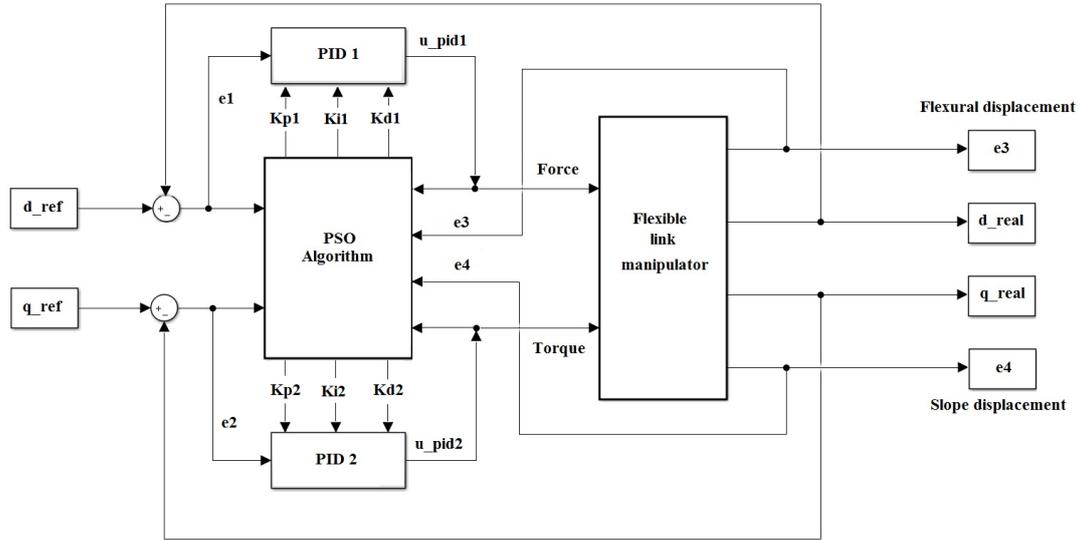


Fig. 4. Structural control in MATLAB/SIMULINK.

From fig. 4, the objective is to tune the PID parameters with minimum consumable energy and minimum errors which are joints variables

$$\begin{aligned} e_1 &= d\_ref - d\_real \\ e_2 &= q\_ref - q\_real \end{aligned} \tag{31}$$

Where,  $d\_ref$  and  $q\_ref$  are the reference points or desire path. Symbols  $d\_real$  and  $q\_real$  are values of joints variables which are controlled. Errors  $e_3$  and  $e_4$  are elastic displacements at the end-effector point of flexible manipulator. Symbol  $u\_pid1$  and  $u\_pid2$  are driving force and torque which are PID control laws. Parameters  $K_{p1}, K_{i1}, K_{d1}$  and  $K_{p2}, K_{i2}, K_{d2}$  are proportional gain, integral, derivative times of controllers, respectively. With  $T_d$  is the control time and defining vectors

$$e = [e_1 \ e_2 \ e_3 \ e_4] \text{ and } u = [u_{pid1} \ u_{pid2}] , \text{ the objective function } J = \int_0^{T_d} (e^T e + u^T u) dt \text{ is used in}$$

PSO. Fitness function  $J$  is the linear quadratic regulator (LQR) function. Function  $J$  includes the sum-squared of error between the desire output  $d\_ref$  which produced from the input to the system and actual output  $d\_real$  of the system and sum-squared of driving energy. The optimum target is finding the minimum cost of  $J$  function with values of respective parameters of PID controllers which are changed from lower bound to upper bound values.

#### 4. Simulation results

In this work, simulation results are presented for two cases. Case 1 is position control and case 2 is path control in joint space. Parameters of dynamic model, reference point and desire path are shown in Table. 1.

Table. 1. Parameters of dynamic model

Property	Symbol	Value
Length of link 1 (m)	$L_1$	0.1
Mass of link 1 and base (kg)	$m_1$	1.4
Parameters of link 2		
Length of link (m)	$L_2$	0.3
Width (m)	$b$	0.02
Thickness (m)	$h$	0.001
Number of element	$n$	5
Cross section area (m <sup>2</sup> )	$A=b.h$	$2.10^{-5}$
Mass density (kg/m <sup>3</sup> )	$\rho$	7850
Mass per meter (kg/m)	$m=\rho.A$	0.157
Young's modulus (N/m <sup>2</sup> )	$E$	$2.10^{10}$
Inertial moment of cross section (m <sup>4</sup> )	$I=b.h^3/12$	$1.67 \times 10^{-12}$
Damping ratios	$\alpha, \beta$	0.005; 0.07
Mass of payload (g)	$m_t$	10
Reference values of translational joint (m)	$d_{ref}$	0.2
Reference values of rotational joint (rad)	$q_{ref}$	1.57
Desire path of translational joint (m)	$d(t)_{ref}$	$\sin(t)$
Desire path of rotational joint (rad)	$q(t)_{ref}$	$\sin\left(\frac{\pi}{4}t\right)$
Time simulation (s)	$T$	10

Parameters are used in PSO following Table. 2.

Table. 2. Parameters of PSO algorithm for two cases control

Property	Value
Number of particles in a swarm	50
Number of searching steps for a particle	20
Cognitive and social acceleration	2
Max and min inertia factor	0.9; 0.4
Number of optimization variables	6
Lower bound of variables	0
Upper bound of variables	30

It noted that values of lower and upper bound of variables are determined from auto tuning mode in MATLAB/SIMULINK. The optimum parameters in this case are shown in Table 3.

Table 3. Simulation results

Property	Value
Case 1- Position control in joint space	
Optimum values ( $K_p$ , $K_i$ , $K_d$ ) of PID 1	22.5673; 0.357; 5.517
Optimum values ( $K_p$ , $K_i$ , $K_d$ ) of PID 2	29.056; 0.544; 5.017
Case 2- Path control in joint space	
Optimum values ( $K_p$ , $K_i$ , $K_d$ ) of PID 1	29.08; 0.247; 16.59
Optimum values ( $K_p$ , $K_i$ , $K_d$ ) of PID 2	18.85; 0.032; 18.775

Simulation results in Case 1 are presented in fig. 5, fig. 6 and fig. 7. The simulation results of translational joint, error of joint are shown in fig. 5. Considering translational joint value, rise time is 0.6(s). Settling time is 3(s), maximum overshoot is 20(%) at 0.8(s) but it reduces fast after that, state error is zero.

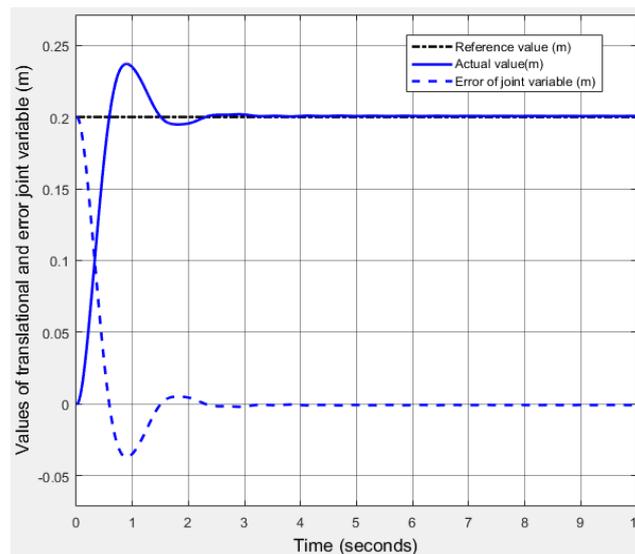


Fig. 5. Values of translational joint and error of joint variable.

Maximum error of translational joint variable is 0.04(mm). Value of rotational and error of joint variable between reference and actual value are show in fig. 6. Rise time is 0.5(s), settling time is 1(s), overshoot value is zero and state error value is zero, too. The elastic displacements at the end-effector point are reduced and show in fig. 7. Maximum value of flexural displacement is 0.12(m) at 0.5(s) and reduces fast. Maximum value of slope displacement is 0.62(rad) at 0.5(s) and reduces very fast than reducing of flexural displacement value.

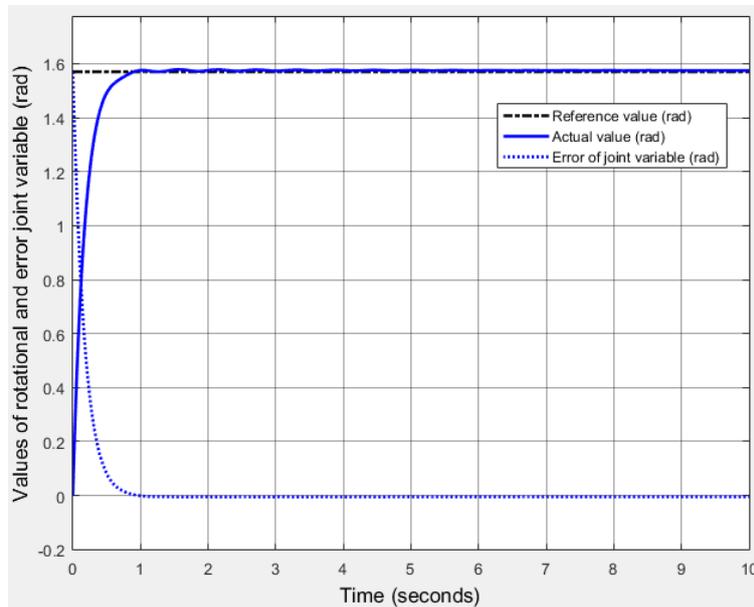


Fig. 6. Values of rotational joint and error of joint variable.

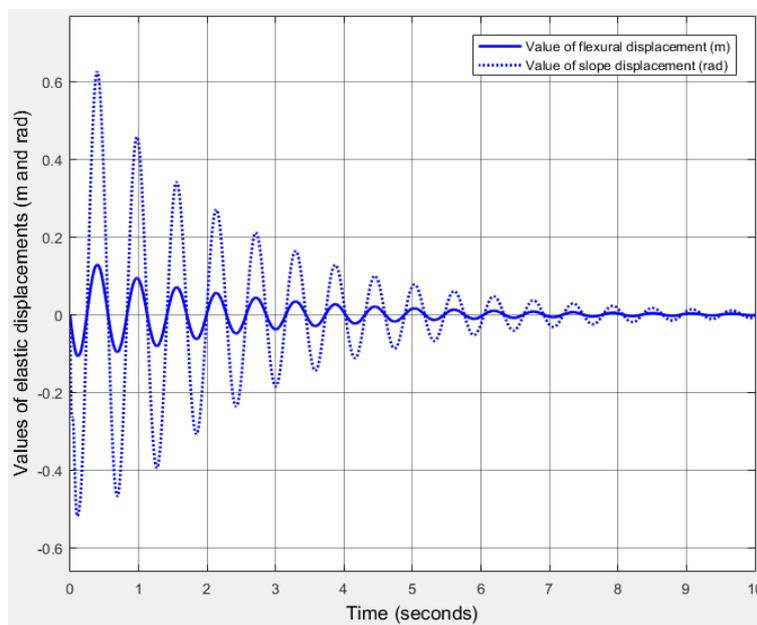


Fig. 7. Values of elastic displacements at end-effector point.

Simulation results in Case 2 are presented from fig. 8 to fig. 11.

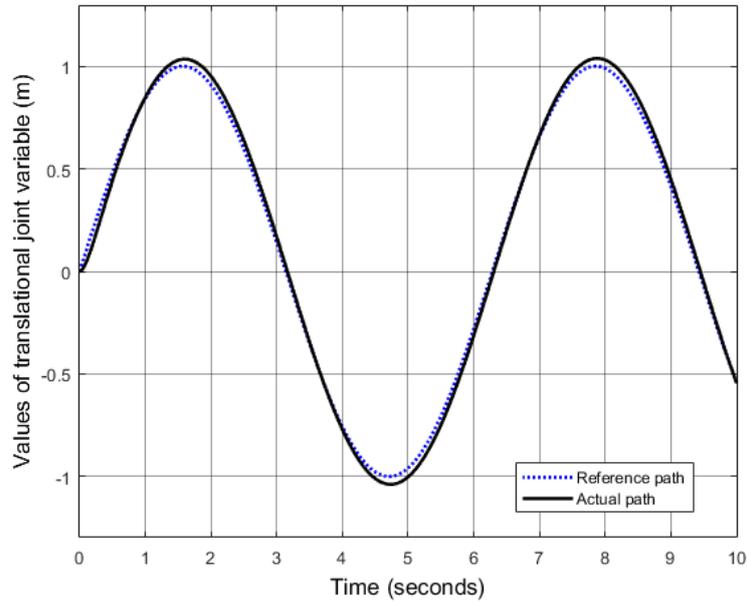


Fig. 8. Control result for translational joint

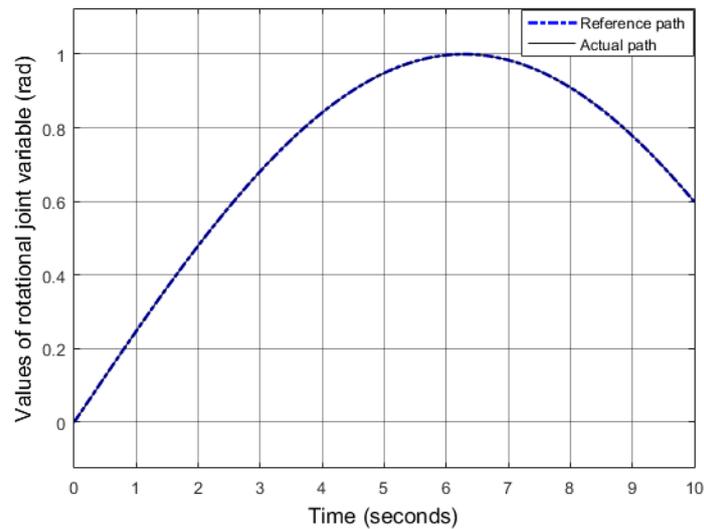


Fig. 9. Control result for rotational joint.

Maximum flexural displacement is 3.8(mm). This value reduced to displacement value at static state after 1.4(s). The maximum slope displacement is 0.175(rad). It reduced to displacement value at static state after 1.4(s) too. The velocities of elastic displacements are shown in fig. 10 and fig. 11.

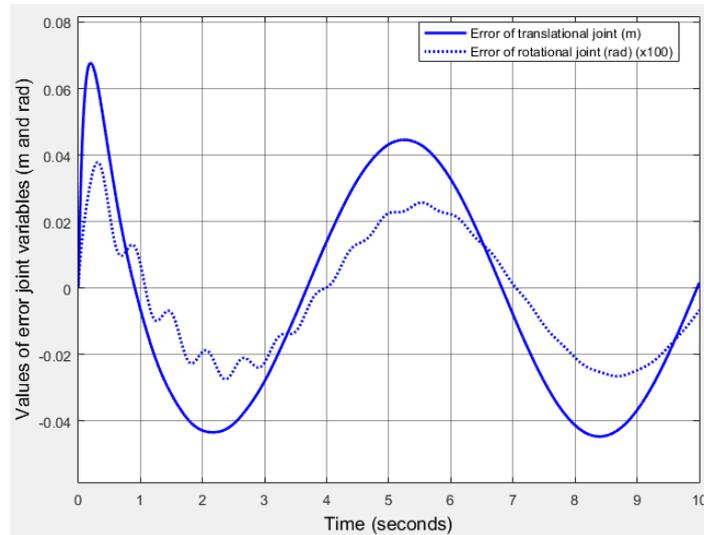


Fig. 10. Errors of joint variables.

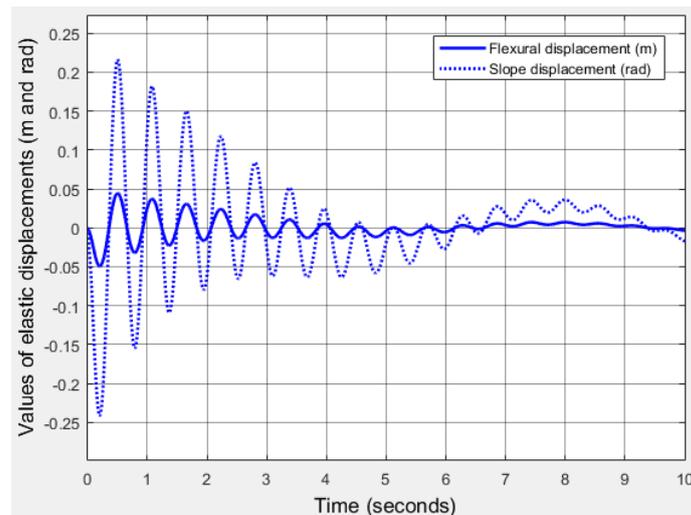


Fig. 11. Values of elastic displacements at the end-effector point.

Based on control results in fig. 8 and fig. 9, the control quality is high efficiency with small errors which are shown in fig. 10. Elastic displacements in fig. 11 are smaller than these displacements in case 1. Maximum values of flexural and slope displacement are 0.05(mm) and 0.22(rad) at 0.5(s), respectively.

In general, simulation results show that initial control requests are warranted. The errors of joint variables are small and fast response. However, elastic displacements are not absolutely eliminated and these values effect on position of end-effector point in workspace. This problem will be considered and solved in the next paper.

## 5. Conclusion

A nonlinear dynamic model of a flexible link robot with rigid translational and rotational joints is presented. Equations of motion are built based on using finite element method and Lagrange approach. PID control system is proposed to warrant following reference point and desire path in joint space based on errors of joint variables and value of elastic displacement at the end-effector point. Parameters of PID control are optimized by using PSO algorithm. The output search results are successfully applied to control position. The approach method and results of this study can be referenced to research other flexible robot with more joint or other joint styles. There are remaining issues which need be studied further in future work.

## References

- [1] P. K. C Wang and Jin Duo Wei, Vibration in a moving flexible robot arm, *Journal of Sound and vibration* 116 (1987) 149.
- [2] P. K. C Wang and Jin Duo Wei, Feedback control of vibrations in a moving flexible robot arm with rotary and prismatic joints, *Proceedings of the IEEE International Conference on Robotics and Automation*, Raleigh, North Carolina, March (1987) 1683.
- [3] D. S. Kwon and W. J. Book, A time-domain inverse dynamic tracking control of a single link flexible manipulator, *Journal of Dynamic Systems, Measurement and Control* 116 (1994) 193.
- [4] Pan. Y. C, Scott and R. A. Ulsoy, Dynamic modeling and simulation of flexible robots with translational joints, *J. Mech. Design* 112 (1990) 307.
- [5] Yuh. J and Young. T, Dynamic modeling of an axially moving beam in rotation: simulation and experiment, *Trans. ASME J. Dyn. Syst. Meas. Control* 113 (1991) 34.
- [6] Al-Bedoor. B. O and Khulief. Y. A, General planar dynamics of a sliding flexible link, *Sound and Vibration* 206 (1997) 641.
- [7] S. E. Khadem and A. A. Pirmohammadi, Analytical development of dynamic equations of motion for a three-dimensional flexible manipulator with revolute and prismatic joints, *IEEE Trans. Syst. Man Cybern. B Cybern* 33 (2003) 237.
- [8] M. H. Korayem, A. M. Shafei and S. F. Dehkordi, Systematic modeling of a chain of N-flexible link manipulators connected by revolute–prismatic joints using recursive Gibbs-Appell formulation, Springer-Verlag, Berlin Heidelberg, 2014.
- [9] S. S. Gee, T. H. Lee and G. Zhu, A Nonlinear feedback controller for a single link flexible manipulator based on a finite element method, *Journal of robotics system* 14 (1997) 165.
- [10] Kuo. Y. K and J. Lin, Fuzzy logic control for flexible link robot arm by singular perturbation approach, *Applied Soft Computing* 2 (2002) 24.
- [11] Tang Yuan-Gang, Fu-Chun Sun and Ting-Liang Hu, Tip Position Control of a Flexible-Link Manipulator with Neural networks, *International Journal of control automation and systems* 4 (2006) 308.
- [12] Yatim. H. M and I. Z. Mat Darus, *Swarm Optimization of an Active Vibration Controller for Flexible*, Control and Signal Processing, ISBN: 978-1-61804-173-9 (2010).
- [13] Huang. J. W, Jung-Shan Lin, Back-stepping Control Design of a Single-Link Flexible Robotic Manipulator, *Proceedings of the 17th World Congress The International Federation of Automatic Control*, Seoul, Korea (2008).
- [14] Tokhi. M. O, A. K. M. Azad, *Flexible robot manipulators (modeling, simulation and control)*, The Institution of Engineering and Technology, London, United Kingdom, ISBN: 978-0-86341-448-0 (2008).
- [15] Kennedy. J and R. Eberhart, Particle Swarm Optimization, *Proceedings of IEEE International Conference on Neural Networks*, Perth, (1995) 1942.