

## BACKSTEPPING METHOD FOR A SINGLE-LINK FLEXIBLE-JOINT MANIPULATOR USING GENETIC ALGORITHM

ALI REZA SAHAB<sup>1</sup> AND MOHAMMAD REZA MODABBERNIA<sup>2</sup>

<sup>1</sup>Department of Engineering, Electrical Engineering Group  
Islamic Azad University, Lahijan Branch  
Shaghayegh Street, Lahijan, Guilan, Iran  
sahab@iau-lahijan.ac.ir

<sup>2</sup>Department of Electronics  
Chamran Community Collage of Technology  
Rasht, Iran  
m\_modabbernia@afr.ac.ir

Received April 2010; revised August 2010

**ABSTRACT.** Flexible manipulators are extensively used in industries. In this paper, backstepping method (BM) is used to control flexible manipulator. BM consists of parameters which accept positive values. The parameters are usually chosen variously. The system responses differently for each value. In this method, some parameters exist, which, if not defined well, may cause some performance degrade. It is necessary to select proper parameters to obtain a good response because the improper selection of the parameters leads to inappropriate responses or even may lead to instability of the system. Genetic algorithms (GA) are used to compute the optimal parameters for the backstepping controller of single-link flexible-joint manipulator systems. GA can select appropriate and optimal values for the parameters. GA minimize the fitness function, so the optimal values for the parameters will be found. Selected fitness function is defined to minimize the least square error. Fitness function enforces the system error to decay to zero rapidly. Hence, it causes the system to have a short and optimal setting time. Fitness function also makes an optimal controller and causes overshoot to reach to its minimum value. This hybrid leads to optimal backstepping controller (OBM).

**Keywords:** Single-link flexible-joint manipulator, Lyapunov function, Backstepping method, Genetic algorithm

**1. Introduction.** Industrial manipulator robots play an important role in the field of flexible automation. A single link manipulator is the most basic one which is operated to perform tasks such as moving payloads or painting objects. To obtain a high performance single link manipulator, position controllers are necessary in order to follow a preselected positional trajectory specified either as point-to-point or continuous path tracking motion with minimal deviation by manipulator.

In recent years of considering flexible manipulator, several papers were published. Most of them considered following items: 1. The necessity of considering flexibility. 2. Flexible manipulator modeling. 3. Simple controller design. 4. Analysis of the flexible manipulator specifications such as controllability. Many investigators worked to control the position of the end-effector of the single-link manipulators. Here in this section, the methods used by different investigators are described very briefly, while the computed torque control [1] inversion based on control schemes [2] for the end-point control of single-link flexible manipulator. Adaptive control schemes [3] for tip position control of single-link manipulators. Used robust control schemes for the single-link manipulator [4]. Lyapunov based

control was proposed [5-7]. Slewing control of single-link manipulators was studied [8]. The sensor based feedback controls were carried out [9], adaptive sliding control [10] and robust linear controller [11] for single-link manipulators.

Backstepping method (BM) is used in this study to control flexible manipulator. Various methods are used to enhance the backstepping controller and achieve adequate performance such as some modern control techniques which are used [12,13]. Some optimization techniques are used to optimize the performance [14,15]. Smart material is used to control the vibration of flexible manipulators [16]. PD controllers and PZT actuators control the vibration of the single-link flexible manipulator [17].

One of the most useful optimization method is genetic algorithms (GA). GA have been extensively applied to the off-line design of controllers [18]. In [19], GA were used to design a multi-modal command shaper to reduce end-point vibration of a single-link flexible manipulator having many resonance modes associated with different damping ratios. Heuristically selected weighted sum of multiple competing objectives were used to reduce vibration as well as achieve satisfactory response. The success of this approach depends on suitable selection of weight vector that requires prior knowledge of the system. Moreover, several design goals cannot be guaranteed in this method.

BM in practice causes delay in response to the system while it reduces vibration and the amount of reduction in vibration and the rise time is found to be in conflict with another in most flexible systems. Moreover, system performance indicators such as overshoot, rise time, settling time, end-point vibration are found to be in conflict with another for a desired level of performance due to the construction and mode of operation of any flexible systems and manipulators. Although a large amount of work has been done on BM both on design techniques and issues relating to practical applications, very few work is reported to address the conflicting features in response to the system and user's demand while BM is designed for a single-link flexible manipulator. The main contribution of this paper is to design a BM using GA optimization to control a single-link flexible manipulator.

**2. Modeling and Problem Formulation.** Giving a priori the motor inertia, it identifies the rest parameters of the system without requiring acceleration signals. The dynamics of a single-link flexible-joint manipulator can be described by [20].

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{mgl}{J_l} \sin x_1 - \frac{k_e}{J_l}(x_1 - x_3) \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= \frac{k_e}{J_r}(x_1 - x_3) - \frac{\mu}{J_r}x_4 + \frac{1}{J_r}u(t) \end{aligned} \tag{1}$$

where  $x_1$  is the link position,  $x_2$  is the link angular velocity,  $x_3$  is the motor rotor position,  $x_4$  is the motor rotor angular velocity,  $J_l$  is the link inertia,  $J_r$  is the motor rotor inertia,  $k_e$  is the joint elastic constant,  $m$  is the link mass,  $l$  is the link length,  $g$  is the gravity constant,  $\mu$  is the viscosity, and  $u(t)$  is the control input. The system parameters are chosen from [21] that list in Table 1.

**3. Backstepping Method.** Considering the strict-feedback nonlinear system as follow:

$$\begin{aligned} \dot{x}_i &= f_i(x_1, \dots, x_i) + g_i(x_1, \dots, x_i)x_{i+1} \\ \dot{x}_n &= f_n(x) + g_n(x)u; \quad 1 \leq i \leq n-1 \end{aligned} \tag{2}$$

where  $x = [x_1, \dots, x_n]^T$ ,  $f_i(0)$  and  $g_i(0)$  are functions with  $f_i(0) = 0$  and  $g_i(0) \neq 0$ .

TABLE 1. Parameters of system

Symbols	Values	Units
$mgl$	5	[N m]
$J_l$	1	[Kg m <sup>2</sup> ]
$J_r$	0.3	[Kg m <sup>2</sup> ]
$\mu$	0.1	[Kg m <sup>2</sup> /sec]
$k_e$	100	[N m]

**Step 1.** Considering the first subsystem of (2),  $x_2$  is taken as a control input and choose.

$$\dot{x}_2 = \frac{1}{g_1(x_1)}[u_1 - f_1(x_1)] \tag{3}$$

The first subsystem is changed to be  $\dot{x}_1 = u_1$ . Choosing  $u_1 = -k_1x_1$  with  $k_1 > 0$ , the origin of the first subsystem  $x_1 = 0$  is asymptotically stable, and the corresponding Lyapunov function is  $V_1(x_1) = x_1^2/2$ , (3) is changed to:

$$x_2 = \varphi_1(x_1) = \frac{1}{g_1(x_1)}[-k_1x_1 - f_1(x_1)] \tag{4}$$

**Step 2.** Take  $x_3$  as a virtual control and the  $(x_1, x_2)$  subsystem is changed to (6).

$$x_3 = \frac{1}{g_1(x_1, x_2)}[u_2 - f_2(x_1, x_2)] \tag{5}$$

$$\begin{aligned} \dot{x}_1 &= f_1(x_1) + g_1(x_1)x_2 \\ \dot{x}_2 &= u_2 \end{aligned} \tag{6}$$

which is in the form of BM, so the control law  $u_2$  is as follow:

$$u_2 = -\frac{\partial V_1}{\partial x_1}g_1(x_1) - k_2[x_2 - \varphi_1(x_1)] + \frac{\partial \varphi_1}{\partial x_1}[f_1(x_1) + g_1(x_1)x_2] \tag{7}$$

where  $k_2 > 0$ . This control law asymptotically stabilizes  $(x_1, x_2) = (0, 0)$  and Lyapunov function is as (8).

$$V_2(x_1, x_2) = V_1(x_1) + \frac{1}{2}[x_2 - \varphi_1(x_1)]^2 \tag{8}$$

Substituting (7) into (5) gives

$$x_3 = \varphi_2(x_1, x_2) = \frac{1}{g_2} \left[ -\frac{\partial V_1}{\partial x_1}g_1 - k_2(x_2 - \varphi_1) + \frac{\partial \varphi_1}{\partial x_1}(f_1 + g_1x_2) - f_2 \right] \tag{9}$$

Until Step  $n$ ,  $u = \varphi_n(x)$  shall be determined which can asymptotically stabilize (2). BM expanded for a class of nonlinear MIMO systems [22].

**4. Controlling Manipulator System.** BM is used to set states  $x_1, x_2, x_3, x_4$  to the origin point  $(0, 0, 0, 0)$  via the torque  $u$  calculated with four steps.

**Step 1.**  $x_2$  is taken as (12) to construct the joint Lyapunov function (11) for (10).

$$\dot{x}_1 = x_2 \tag{10}$$

$$V_0(x_1) = \frac{1}{2}x_1^2 \tag{11}$$

$$x_2 = \varphi_0(x_1) = -k_1x_1 \tag{12}$$

**Step 2.** By considering  $(x_1, x_2)$  of (1) and taking  $x_3$  as a virtual control input and choose.

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{mgl}{J_l} \sin x_1 - \frac{k_e}{J_r} (x_1 - x_3)\end{aligned}\quad (13)$$

$$x_3 = \varphi_1(x_1, x_2) = \frac{J_l}{k_e} \left[ \frac{\partial \varphi_0}{\partial x_1} \dot{x}_1 - \frac{\partial V_0}{\partial x_1} - k_2(x_2 - \varphi_0) + \frac{mgl}{J_l} \sin x_1 + \frac{k_e}{J_l} x_1 \right] \quad (14)$$

$$V_1(x_1, x_2) = V_0 + \frac{1}{2}(x_2 - \varphi_0)^2 \quad (15)$$

**Step 3.** Take virtual control input (17) and Lyapunov function (18) for  $(x_1, x_2, x_3)$  of (16).

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{mgl}{J_l} \sin x_1 - \frac{k_e}{J_l} (x_1 - x_3) \\ \dot{x}_3 &= x_4\end{aligned}\quad (16)$$

$$x_4 = \varphi_2(x_1, x_2, x_3) = \frac{\partial \varphi_1}{\partial x_1} \dot{x}_1 + \frac{\partial \varphi_1}{\partial x_2} \dot{x}_2 - \frac{k_e}{J_l} \frac{\partial V_1}{\partial x_2} - k_3(x_3 - \varphi_1) \quad (17)$$

$$V_2(x_1, x_2, x_3) = V_1 + \frac{1}{2}(x_3 - \varphi_1)^2 \quad (18)$$

**Step 4.** Final control input and Lyapunov function are given in (20) and (21) for (19).

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{mgl}{J_l} \sin x_1 - \frac{k_e}{J_l} (x_1 - x_3) \\ \dot{x}_3 &= x_4\end{aligned}\quad (19)$$

$$\dot{x}_4 = \frac{k_e}{J_r} (x_1 - x_3) - \frac{\mu}{J_r} x_4 + \frac{1}{J_r} u(t)$$

$$u = J_r \left[ \frac{\partial \varphi_2}{\partial x_1} \dot{x}_1 + \frac{\partial \varphi_2}{\partial x_2} \dot{x}_2 + \frac{\partial \varphi_2}{\partial x_3} \dot{x}_3 - \frac{\partial V_2}{\partial x_3} - k_4(x_4 - \varphi_2) - \frac{k_e}{J_r} (x_1 - x_3) + \frac{\mu}{J_r} x_4 \right] \quad (20)$$

$$V_3(x_1, x_2, x_3, x_4) = V_2 + \frac{1}{2}(x_4 - \varphi_2)^2 \quad (21)$$

**5. Tracking Desired Trajectory.** In this section, a control law  $u$  will be found so that a scalar output  $x_1(t)$  of flexible link manipulator can track any desired trajectory. Let  $\bar{x}_1$  be the deviation between the output  $x_1$  and the desired trajectory  $r(t)$ .  $\bar{x}_1 = x_1 - r(t)$ . Therefore, (1) would be converted to (22), as follows. BM is used to bring the state  $\bar{x}_1$  to tracking the desired trajectory  $r(t)$  via the torque  $u$  calculated with four steps.

$$\begin{aligned}\dot{\bar{x}}_1 &= x_2 - \dot{r} \\ \dot{x}_2 &= -\frac{mgl}{J_l} \sin(\bar{x}_1 + r) - \frac{k_e}{J_l} (\bar{x}_1 + r - x_3) \\ \dot{x}_3 &= x_4\end{aligned}\quad (22)$$

$$\dot{x}_4 = \frac{k_e}{J_r} (\bar{x}_1 + r - x_3) - \frac{\mu}{J_r} x_4 + \frac{1}{J_r} u(t)$$

**Step 1.** Take  $x_2$  as (25) input to construct the joint Lyapunov function (24) for (23).

$$\dot{\bar{x}}_1 = x_2 - \dot{r} \quad (23)$$

$$V_0(\bar{x}_1) = \frac{1}{2}\bar{x}_1^2 \tag{24}$$

$$x_2 = \varphi_0(\bar{x}_1) = -k_1\bar{x}_1 \tag{25}$$

**Step 2.** Take  $x_3$  and Lyapunov function according to (27) and (28) for  $(\bar{x}_1, x_2)$  of (22).

$$\begin{aligned} \dot{\bar{x}}_1 &= x_2 - \dot{r} \\ \dot{x}_2 &= -\frac{mgl}{J_l} \sin(\bar{x}_1 + r) - \frac{k_e}{J_l}(\bar{x}_1 + r - x_3) \end{aligned} \tag{26}$$

$$x_3 = \varphi_1(\bar{x}_1, x_2) = \frac{J_l}{k_e} \left[ \frac{\partial \varphi_0}{\partial \bar{x}_1} \dot{\bar{x}}_1 - \frac{\partial V_0}{\partial \bar{x}_1} - k_2(x_2 - \varphi_0) + \frac{mgl}{J_l} \sin(\bar{x}_1 + r) + \frac{k_e}{J_l}(\bar{x}_1 + r) \right] \tag{27}$$

$$V_1(\bar{x}_1, x_2) = V_0 + \frac{1}{2}(x_2 - \varphi_0)^2 \tag{28}$$

**Step 3.** Considering  $(\bar{x}_1, x_2, x_3)$  of (22) take  $x_4$  as a control input and Lyapunov function according to (30) and (31).

$$\begin{aligned} \dot{\bar{x}}_1 &= x_2 - \dot{r} \\ \dot{x}_2 &= -\frac{mgl}{J_l} \sin(\bar{x}_1 + r) - \frac{k_e}{J_l}(\bar{x}_1 + r - x_3) \end{aligned} \tag{29}$$

$$\dot{x}_3 = x_4$$

$$x_4 = \varphi_2(\bar{x}_1, x_2, x_3) = \frac{\partial \varphi_1}{\partial \bar{x}_1} \dot{\bar{x}}_1 + \frac{\partial \varphi_1}{\partial x_2} \dot{x}_2 - \frac{k_e}{J_l} \frac{\partial V_1}{\partial x_2} - k_3(x_3 - \varphi_1) \tag{30}$$

$$V_2(\bar{x}_1, x_2, x_3) = V_1 + \frac{1}{2}(x_3 - \varphi_1)^2 \tag{31}$$

**Step 4.** Final actual control input  $u$  and Lyapunov function are (33) and (34).

$$\begin{aligned} \dot{\bar{x}}_1 &= x_2 - \dot{r} \\ \dot{x}_2 &= -\frac{mgl}{J_l} \sin(\bar{x}_1 + r) - \frac{k_e}{J_l}(\bar{x}_1 + r - x_3) \end{aligned} \tag{32}$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = \frac{k_e}{J_r}(\bar{x}_1 + r - x_3) - \frac{\mu}{J_r}x_4 + \frac{1}{J_r}u(t)$$

$$u = J_r \left[ \frac{\partial \varphi_2}{\partial \bar{x}_1} \dot{\bar{x}}_1 + \frac{\partial \varphi_2}{\partial x_2} \dot{x}_2 + \frac{\partial \varphi_2}{\partial x_3} \dot{x}_3 - \frac{\partial V_2}{\partial x_3} - k_4(x_4 - \varphi_2) - \frac{k_e}{J_r}(\bar{x}_1 + r - x_3) + \frac{\mu}{J_r}x_4 \right] \tag{33}$$

$$V_3(\bar{x}_1, x_2, x_3, x_4) = V_2 + \frac{1}{2}(x_4 - \varphi_2)^2 \tag{34}$$

**6. Genetic Algorithm.** Most of the optimization algorithms are based on the gradient of the cost function, so for the ill choice of the initial point or the interval search, these algorithms can be misled on the local optimum and cannot achieve the global optimum. To solve this problem, a class of optimization algorithms; like genetic algorithms (GA), are developed.

In its most general usage, GA refer to a family of computational models inspired by evolution. These algorithms start with many initial points in order to cover all search intervals and encode a potential solution to a specific problem on a simple chromosome like data structure and apply recombination operators to these structures so as to preserve critical information. An implantation of GA begins with a population of chromosomes randomly bred. Each chromosome is evaluated by using the objective function called fitness function. In order to apply the GA reproductive operations called crossover and

mutation two individuals are selected randomly called parents and apply the crossover operation if its probability reaches between parents by exchanging some of their bits to produce two children. A mutation is the second operator applied on the single children by inverting its bit if the probability reaches. After this stage we obtain two population: parents and children, the individual who has a good solution is preserved [23].

GA is used to search the optimal parameter ( $k$ ) in order to guarantee the stability of systems by ensuring negativity of the Lyapunov function and a suitable time response. (20) is optimized by the fitness function in (35).

$$f(x_1, x_2, \dots, x_n) = \frac{1}{n} \sqrt{\sum_{i=1}^n \int (x_i(t) - x_{di})^2 dt} \quad (35)$$

$x_i$  is the system state and  $x_{di}$  is the favorite mood for  $x_i$ . Based on the system purpose for placing the states at zero value;  $x_{di} = 0$ . (33) is optimized by the fitness function in (36) where  $y(t)$  is system output and  $r(t)$  is reference input.

$$f(y_1, y_2, \dots, y_n) = \frac{1}{n} \sqrt{\sum_{i=1}^n \int (y_i(t) - r(t))^2 dt} \quad (36)$$

**7. Numerical Simulations.** This section presents numerical simulations flexible link manipulator. The Optimal Backstepping Method (OBM) is used as an approach to control manipulator system and eventually the results of this method would be compared with the control result of Robust Control Method (RCM) [24]. The simulations are given in the following four cases.

- Case 1 : Stabilization to the Origin Point (0, 0, 0, 0).
- Case 2 : Tracking Step Input  $r(t) = 1$ .
- Case 3 : Tracking Reference Input  $r(t) = \sin(t)$ .
- Case 4 : Tracking Reference Input  $r(t) = 2 - 2e^{-t}$ .

The optimal parameters of BM are listed in Table 3.

Figures 1 – 4 show that  $x_1$ ,  $x_2$ ,  $x_3$  and  $x_4$  of manipulator system can be stabilized with (20) to the origin point (0, 0, 0, 0) respectively.

TABLE 2. Genetic algorithm parameters

Parameters	Values
Size population	100
Maximum of generation	100
Prob.crossover	75
Prob.mutation	0.001
$k$ Search interval de	[0.1 10]

TABLE 3. Optimal parameters of BM

	$k_1$	$k_2$	$k_3$	$k_4$
Case 1	1.805	4.492	9.444	1.09
Case 2	9.987	9.908	9.985	9.838
Case 3	9.788	0.115	0.165	9.883
Case 4	9.661	9.904	9.992	9.986

Figures 5 – 7 show that the scalar output link position can track the step input and desired trajectory  $r(t) = \sin t$  or  $r(t) = 2 - 2e^{-t}$  with (33).

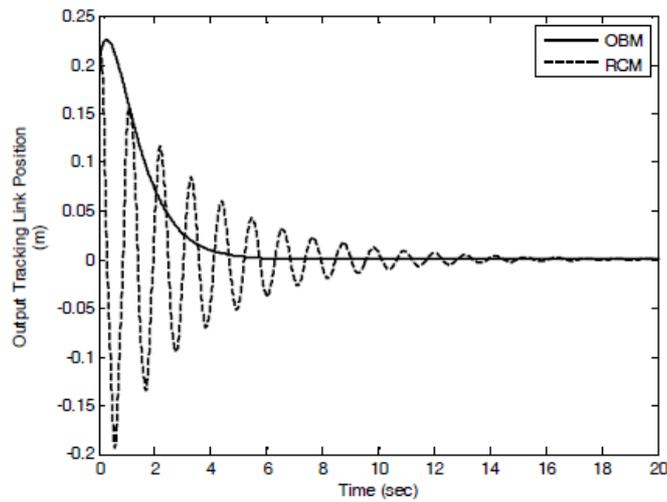


FIGURE 1. Link position

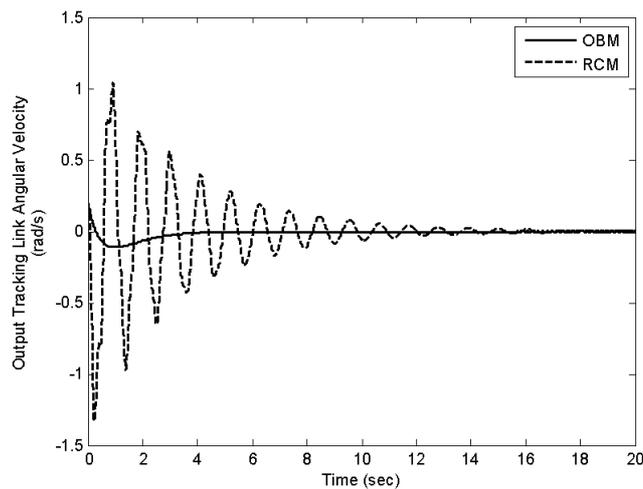


FIGURE 2. Link angular velocity

8. **Conclusion.** A single link manipulator is the most manipulators are extensively used in industries. In this paper, a single-link flexible-joint manipulator is controlled with BM. The designed controller consists of parameters which accept positive values. The controlled system presents different behavior for different values of these parameters. Improper selection of the parameters causes an improper behavior which may cause serious problems such as instability of system. It is needed to optimize these parameters. One of well known optimization method is GA. GA optimize the controller to gain optimal and proper values for the parameters. For this reason, GA minimize the fitness function to find minimum current value for it. On the other hand, fitness function finds minimum value to minimize least square errors. Simulation results show that the setting time and overshoot, reach to their minimum values that is demonstrated to have more optimal values when compared with previous methods. Also for different control goal performance different fitness function can be selected to have other appropriate results.

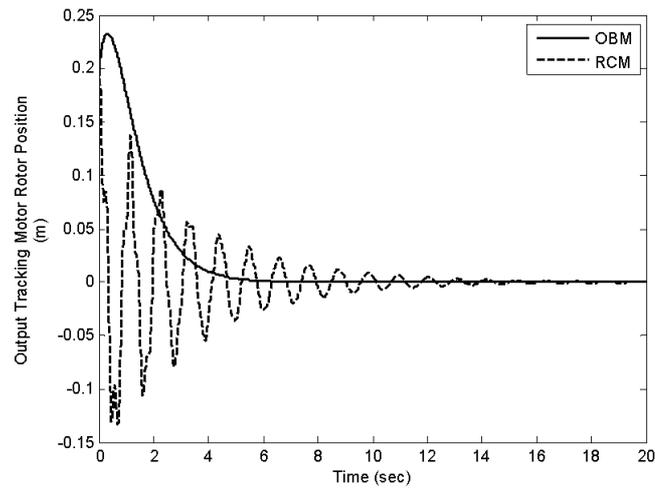


FIGURE 3. Motor rotor position

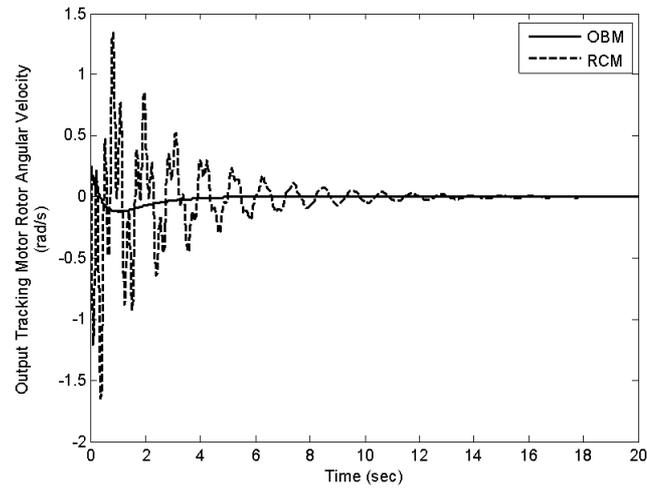


FIGURE 4. Motor rotor angular velocity

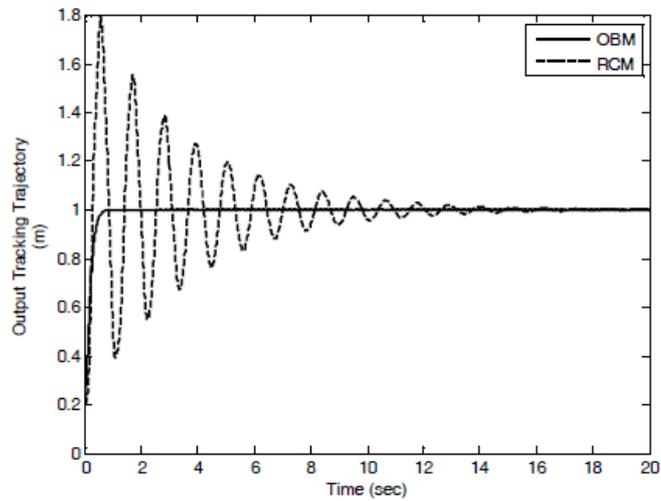
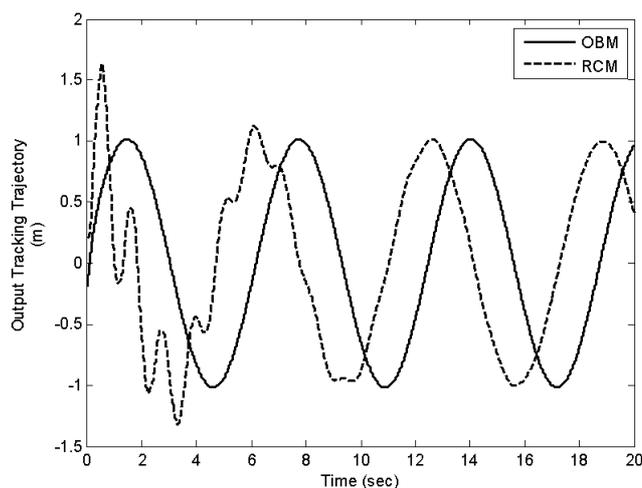
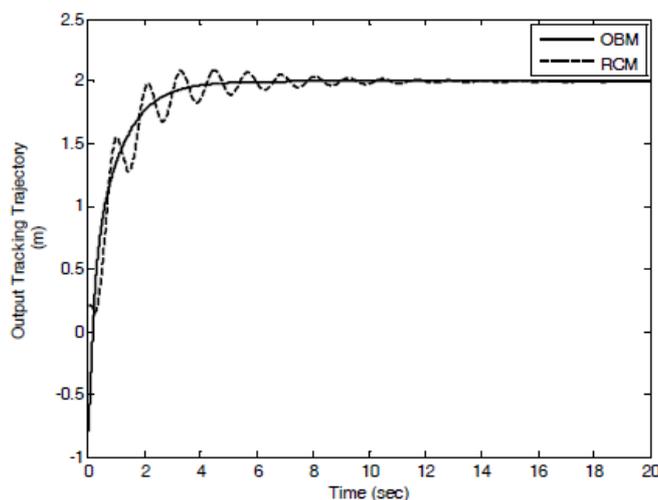


FIGURE 5. Track the step input

FIGURE 6. Track  $r(t) = \sin t$ FIGURE 7. Track  $r(t) = 2 - 2e^{-t}$ 

## REFERENCES

- [1] L. Y. Liu and K. Yuan, Noncollocated passivity-based PD control of a single-link flexible manipulator, *Robotica*, vol.21, pp.117-135, 2003.
- [2] K. S. Rattan, V. Feliu and H. B. Brown, Identification and control of a single-flexible manipulator, *Proc. of the IEEE International Conference on Robotics and Automation*, vol.3, pp.2278-2279, 1988.
- [3] J. J. Feliu, V. Feliu and C. Cerrada, Load adaptive control of single link flexible arms based on new modeling technique, *IEEE Trans. on Robotics and Automation*, vol.15, no.(5), pp.793-804, 1999.
- [4] V. Feliu and F. Ramos, Strain gauge based control of single-link flexible very light weight robots robust to payload changes, *Mechatronics*, vol.15, pp.547-571, 2005.
- [5] C. Yong and B. L. Walcott, Control of a one-link flexible manipulator via Lyapunov methods, *IEEE International Conference on System Engineering*, pp.575-579, 1989.
- [6] J. J. Shiffman, Lyapunov functions and the control of the euler-bernoulli beam, *International Journal of Control*, vol.57, no.4, pp.971-990, 1993.
- [7] M. Dadfarnia, N. Jalili, B. Xian and D. M. Dawson, A Lyapunov-based piezoelectric controller for flexible Cartesian robot manipulator, *ASME Journal of Dynamic Systems, Measurement, and Control*, vol.126, pp.347-358, 2004.
- [8] J. Shan, H. T. Liu and D. Sun, Slewing and vibration control of a single-link flexible manipulator by positive position feedback (PPF), *Mechatronics*, vol.15, pp.487-503, 2005.

- [9] S. B. Choi, S. S. Cho, H. C. Shin and H. K. Kim, Quantitative feedback theory control of a single-link flexible manipulator featuring piezoelectric actuator and sensor, *Smart Material Structures*, vol.8, pp.338-349, 1999.
- [10] A. C. Huang and Y. C. Chen, Adaptive sliding control for single-link flexible-joint robot with mismatched uncertainties, *IEEE Trans. on Contr. Syst. Technol.*, vol.12, no.5, pp.770-776, 2004.
- [11] H. D. Taghirad and M. A. Khosravi, A robust linear controller for flexible joint manipulators, *Proc. of IEEE/RSJ Int. Conf. Intell. Rob. Syst., IROS'04*, vol.3, pp.2936-2941, 2004.
- [12] T. Wang, S. Tong and Y. Li, Robust adaptive fuzzy control for nonlinear system with dynamic uncertainties, based on backstepping, *International Journal of Innovative Computing, Information and Control*, vol.5, no.9, pp.2675-2688, 2009.
- [13] S. Tong, Y. Li and T. Wang, Adaptive fuzzy backstepping fault-tolerant control for uncertain nonlinear systems based on dynamic surface, *International Journal of Innovative Computing, Information and Control*, vol.5, no.10(A), pp.3249-3262, 2009.
- [14] Y. Wang, Fuzzy clustering analysis by using genetic algorithm, *ICIC Express Letters*, vol.2, no.4, pp.331-337, 2008.
- [15] M. M. Fateh and A. Azarfar, Improving fuzzy control of robot manipulators by increasing scaling factors, *ICIC Express Letters*, vol.3, no.3(A), pp.513-518, 2009.
- [16] S. S. Ge, T. H. Lee, J. Q. Gong and J. X. Xu, Model-free controller design for a single-link flexible smart materials robots, *International Journal of Control*, vol.73, no.6, pp.531-544, 2000.
- [17] D. Sun, J. K. Millsb, J. Shana and S. K. Tso, A PZT actuator control of a single-link flexible manipulator based on linear velocity feedback and actuator placement, *Mechatronics*, vol.14, pp.381-401, 2004.
- [18] A. Chipperfield et al., Multi-objective optimisation in control system design: An evolutionary computing approach, *IFAC World Congress*, Barcelona, 2002.
- [19] M. O. Tokhi et al., Adaptive command shaping using genetic algorithms for vibration control of a single link flexible manipulator, *Proc. of the 12th International Congress on Sound and Vibration*, 2005.
- [20] F. Ghorbel, J. Hung and M. W. Spong, Adaptive control of flexible-joint manipulators, *IEEE Control Systems Mag.*, vol.9, pp.9-13, 1989.
- [21] H. Sira-Ramirez, S. Ahmad and M. Zribi, Dynamical feedback control of robot manipulators with joint flexibility, *IEEE Trans. on Systems Man Cybernet.*, vol.22, pp.736-747, 1992.
- [22] A. R. Sahab and M. H. Zarif, Improve backstepping method to GBM, *World Applied Sciences Journal*, vol.6, no.10, pp.1399-1403, 2009.
- [23] Z. Michalewicz, *Genetic Algorithm + Data Structures = Evolution Programs*, 2nd Edition, Springer Verlag, 1994.
- [24] J. T. Huang, A new approach to parametric identification of a single-link flexible-joint manipulator, *Journal of Intelligent and Robotic Systems*, vol.37, no.3, pp.273-284. 2003.