

# Type-2 Fuzzy Control for a Flexible-joint Robot Using Voltage Control Strategy

Majid Moradi Zirkohi<sup>1</sup>    Mohammad Mehdi Fateh<sup>1</sup>    Mahdi Aliyari Shoorehdeli<sup>2</sup>

<sup>1</sup>Department of Electrical and Robotic Engineering, Shahrood University of Technology, Shahrood 3619995161, Iran

<sup>2</sup>Department of Electrical Engineering, Khaje Nasir Toosi University of Technology, Tehran, Iran

---

**Abstract:** Type-1 fuzzy sets cannot fully handle the uncertainties. To overcome the problem, type-2 fuzzy sets have been proposed. The novelty of this paper is using interval type-2 fuzzy logic controller (IT2FLC) to control a flexible-joint robot with voltage control strategy. In order to take into account the whole robotic system including the dynamics of actuators and the robot manipulator, the voltages of motors are used as inputs of the system. To highlight the capabilities of the control system, a flexible joint robot which is highly nonlinear, heavily coupled and uncertain is used. In addition, to improve the control performance, the parameters of the primary membership functions of IT2FLC are optimized using particle swarm optimization (PSO). A comparative study between the proposed IT2FLC and type-1 fuzzy logic controller (T1FLC) is presented to better assess their respective performance in presence of external disturbance and unmodelled dynamics. Stability analysis is presented and the effectiveness of the proposed control approach is demonstrated by simulations using a two-link flexible-joint robot driven by permanent magnet direct current motors. Simulation results show the superiority of the IT2FLC over the T1FLC in terms of accuracy, robustness and interpretability.

**Keywords:** Type-2 fuzzy controller, flexible-joint robots, voltage control strategy, particle swarm optimization (PSO), actuator.

---

## 1 Introduction

Electrically driven robots are efficiently used in various applications. Because their motors provide low torque at high speed, the robotic systems are equipped with power transmission systems to provide high torque at low speed for performing the tasks. However, deformation of the transmission system produces flexibility in the joints. This phenomenon is the main source of vibration in industrial robot manipulators. Compared with rigid robots, the number of degrees of freedom becomes twice as the number of control actions due to flexibility in the joints, and the matching property between nonlinearities and inputs is lost<sup>[1]</sup>. Performing high-precision tasks by a flexible-joint robot seems to be difficult since the link position cannot follow the actuator position directly. As a result, flexibility in joints should be compensated to improve the performance and avoid unwanted oscillations. However, controlling such systems still faces numerous challenges such as the severe nonlinearities, weak coupling, joint flexibility, varying operating conditions, and wide range of uncertainties<sup>[2]</sup>.

Over the years, researchers attempted various control methods for flexible-joint manipulators including singular perturbation theory<sup>[3]</sup>, feedback linearization<sup>[4]</sup>, adaptive control<sup>[5]</sup>, sliding mode control<sup>[6]</sup>, robust control<sup>[7]</sup>, fuzzy control<sup>[8]</sup> and neural control<sup>[9]</sup>. The presented methods belong to the class of the commonly used control strategy, which is called torque control strategy. The torque control strategy pays attention to dynamics of robot manipulator. However, model of the flexible robot is so complicated,

highly nonlinear, heavily coupled, computationally extensive and uncertain. Thus, torque control approaches, particularly the model-based techniques, face the challenging problems associated with manipulator dynamics<sup>[10]</sup>. It is found that the voltage control strategy<sup>[11]</sup> is superior to the torque-control strategy for the robust control of the rigid manipulators<sup>[12]</sup> in terms of simplicity in the controller design and performance of the control system. Since flexibility in the joints provides complex dynamics, the voltage control strategy is more efficient than the torque-control strategy. Recently, robust control<sup>[10]</sup> and adaptive control<sup>[13]</sup> of flexible-joint robots have been developed using the voltage control strategy. In [14], a neural-network-based adaptive controller has been proposed for the tracking problem of manipulators with uncertain kinematics, dynamics and actuator model, using voltage control strategy.

On another aspect, tools of computational intelligence, such as artificial neural networks and fuzzy logic controllers, have been credited in various applications as powerful tools. They can provide robust controllers for mathematically ill-defined systems subjected to structured and unstructured uncertainties<sup>[15]</sup>. Fuzzy control, as a model-free approach, is simply designed for complicated systems that may be difficult to model analytically<sup>[16, 17]</sup>. Type-1 fuzzy logic control systems (T1FLC) are known for their ability to compensate for structured and unstructured uncertainties to a certain degree. Compared to T1FLC, type-2 fuzzy logic control systems (T2FLC) have been credited to be more powerful in compensating for even higher degrees of uncertainties<sup>[18]</sup>. The concept of type-2 fuzzy set was introduced by Zadeh in 1975 as an extension of the type-1

fuzzy set<sup>[19]</sup>. Type-2 fuzzy sets are able to model such uncertainties because their membership functions (MFs) are fuzzy. MFs of type-1 fuzzy sets are two-dimensional, whereas membership functions of type-2 fuzzy sets are three-dimensional. The third dimension of type-2 fuzzy sets provides an additional degree to model the uncertainties directly<sup>[18]</sup>. In contrast, the performance of a T1FLC may be improved by partitioning the input domains with a larger number of fuzzy sets. Then, a trade-off between accuracy, performance and interpretability is required<sup>[20]</sup>. A larger number of MFs result in a bigger rule base that would be harder for a human to interpret because of the curse of dimensionality<sup>[21]</sup>. The IT2FLC is particularly suitable for time-variant systems with unknown time-varying dynamics<sup>[22]</sup>. Though fuzzy control is the most widely used application of fuzzy set theory, T2FLC have been used in very few control applications, such as nonlinear control and mobile robot navigation<sup>[22]</sup>, decision making<sup>[23]</sup>, and quality control of sound speakers<sup>[24]</sup>.

In order to obtain optimal performance of control systems, optimization algorithms, such as genetic algorithm (GA) and particle swarm optimization (PSO), have been frequently used<sup>[25]</sup>. The PSO was first introduced by Kennedy and Eberhart in 1995<sup>[26]</sup>. Through the simulation of a simplified social system, the behavior of PSO can be treated as an optimization process. As compared with other optimization algorithms like GA, the PSO requires less computational time. Therefore, it has successfully been applied to solve many optimization problems<sup>[25, 27]</sup>.

This paper develops an interval T2FLC (IT2FLC) approach using voltage control strategy to control electrically driven flexible-joint robot manipulators. The IT2FLC is computationally cheaper and more efficient than T2FLC<sup>[18]</sup>.

The novelty is the design of IT2FLC using the voltage control strategy and employing PSO to optimize control design parameters. As a result, the control performance is improved. Stability analysis is presented and the superiority of the IT2FLC to T1FLC is shown in the presence of uncertainties. This paper is organized as follows. Section 2 presents modeling of flexible-joint robots. Interval type-2 fuzzy logic is briefly introduced in Section 3. Section 4 outlines the particle swarm optimization. The design of the proposed controller is detailed in Section 5. Stability analysis is presented in Section 6. Section 7 presents the simulation results. Section 8 concludes the paper.

## 2 Electrically driven flexible-joint robot dynamics

In a simplified model of the flexible-joint robot<sup>[7]</sup>, the manipulator links are assumed rigid and motors are elastically coupled to the links. The motor torques are assumed as inputs of the robotic system. In this paper, the simplified model is applied on an electrically driven robot with some modifications to obtain the motor voltages as the in-

puts. In this study, we consider a two-link manipulator<sup>[28]</sup> as shown in Fig.1 with revolute joints driven by geared permanent magnet direct current motors. If the joint flexibility is modeled by a linear torsional spring, the dynamic equation of motion can be expressed as

$$D(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta) + \tau_f(\dot{\theta}) = K(r\theta_m - \theta) \quad (1)$$

$$J\ddot{\theta}_m + B\dot{\theta}_m + rK(r\theta_m - \theta) = \tau \quad (2)$$

where  $\theta \in \mathbf{R}^2$  is a vector of joint angles, and  $\theta_m \in \mathbf{R}^2$  is a vector of rotor angles. Thus, this system possesses 4 coordinates as  $[\theta \ \theta_m]$ . The matrix  $D(\theta)$  is a  $2 \times 2$  matrix of manipulator inertia,  $C(\theta, \dot{\theta})\dot{\theta} \in \mathbf{R}^2$  is the vector of centrifugal and Coriolis forces,  $g(\theta) \in \mathbf{R}^2$  is a vector of gravitational forces,  $\tau \in \mathbf{R}^2$  is a torque vector of motors, and  $\tau_f(\dot{\theta})$  is the frictional torques. The diagonal matrices  $J, B$  and  $r$  represent coefficients of the motor inertia, motor damping, and reduction gear, respectively. The diagonal matrix  $K$  represents the lumped flexibility provided by the joint and reduction gear. To simplify the model, both the joint stiffness and gear coefficients are assumed constant. The vector of gravitational forces  $g(\theta)$  is assumed the function of only the joint positions as used in the simplified model<sup>[7]</sup>. Note that the vector and matrix are represented in bold form for clarity.

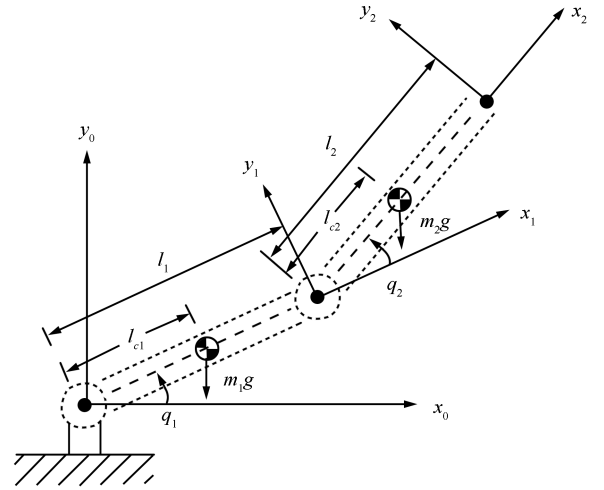


Fig.1 An articulated two-link manipulator

The details are given in [28].

$$\begin{aligned} D(\theta) &= \begin{bmatrix} d_1 & d_2 \\ d_2 & d_3 \end{bmatrix} \\ C(\theta, \dot{\theta}) &= \begin{bmatrix} -m_2 l_1 l_{c_2} \dot{\theta}_2 \sin(\theta_2) & -m_2 l_1 l_{c_2} (\dot{\theta}_1 + \dot{\theta}_2) \sin(\theta_2) \\ m_2 l_1 l_{c_2} \dot{\theta}_1 \sin(\theta_2) & 0 \end{bmatrix} \\ G(\theta) &= \begin{bmatrix} (m_1 l_{c_1} + m_2 l_1) g \cos(\theta_1) + m_2 l_{c_2} g \cos(\theta_1 + \theta_2) \\ m_2 l_{c_2} g \cos(\theta_1 + \theta_2) \end{bmatrix} \\ \tau_f(\dot{\theta}) &= \begin{bmatrix} b_1 \dot{\theta}_1 + c_1 \text{sign}(\dot{\theta}_1) \\ b_2 \dot{\theta}_1 + c_2 \text{sign}(\dot{\theta}_1) \end{bmatrix} \end{aligned} \quad (3)$$

where

$$d_1 = m_1 l_{c_1} + m_2 (l_1^2 + l_{c_1}^2 + 2l_1 l_{c_1} \cos(\theta_2)) + l_1 + l_2 \quad (4)$$

$$d_2 = m_2 (l_{c_1}^2 + l_1 l_{c_1} \cos(\theta_2)) + l_2 \quad (5)$$

$$d_3 = m_2 l_{c_2}^2 + l_2 \quad (6)$$

$\theta_i$  ( $i = 1, 2$ ) denotes the joint angle,  $l_i$  is the link length,  $m_i$  is the link mass,  $I_i$  is the link's moment of inertia given with respect to center of mass,  $l_{c_i}$  is the distance between the center of mass of link, and the  $i$ -th joint  $b_i$  ( $i = 1, 2$ ) and  $c_i$  ( $i = 1, 2$ ) are constants.

System (1)–(2) is highly nonlinear, extensively computational and heavily coupled. Complexity of the model has been a serious challenge in robot modeling and control in literature. It is expected to face a higher complexity if the proposed model includes the actuator dynamics. In order to obtain the motor voltages as inputs, we consider electrical equation of the geared permanent magnet direct current motors in the matrix form

$$u = RI_a + L\dot{I}_a + K_b \dot{\theta}_m \quad (7)$$

where  $u \in \mathbf{R}^2$  is a vector of motor voltages,  $I_a \in \mathbf{R}^2$  is a vector of motor currents, and  $\dot{\theta}_m$  is a vector of rotor velocities. The diagonal matrices  $R, L$  and  $K_b$  represent the coefficients of armature resistance, armature inductance and back-emf constant, respectively. The motor torques  $\tau$ , as input of dynamic equation (2), is produced by the motor currents as

$$K_m I_a = \tau \quad (8)$$

where  $K_m$  is a diagonal matrix of the torque constants. Equalities (1), (2), (7) and (8) form the robotic system such that the voltage vector  $u$  is the input vector and the joint angle vector  $\theta$  is the output vector.

### 3 Interval type-2 fuzzy logic systems

A fuzzy logic system that uses at least one type-2 fuzzy set is called a type-2 fuzzy logic system. It is very useful in circumstances, where the determination of an exact membership grade for a fuzzy set is difficult<sup>[18]</sup>. As illustrated in Fig. 2, a type-2 fuzzy MF can be obtained by starting with a type-1 MF and blurring it. The extra mathematical dimension provided by the blurred area represents the uncertainties in the shape and position of the type-1 fuzzy set and is referred to as the footprint of uncertainty (FOU). The FOU is bounded by upper and lower MFs, and points within the “blurred area” have membership grades given by type-1 MFs. The most frequently used type-2 fuzzy sets are interval fuzzy sets, where each point in the FOU has unity secondary membership grade<sup>[22]</sup>.

An interval type-2 fuzzy set  $\tilde{A}$  in  $X$  is defined as<sup>[19]</sup>

$$\tilde{A} = \int_{x \in X} \left[ \int_{u \in J_x} 1/u \right] /x, \quad J_x \subseteq [0 \ 1] \quad (9)$$

where  $x$  is the primary variable with domain  $X$ ,  $u$  is the secondary variable which has domain  $J_x$ ,  $J_x$  is called the primary membership of  $x$ . Uncertainty about  $\tilde{A}$  is conveyed by the union of all the primary memberships called the footprint of uncertainty (FOU) of  $\tilde{A}$ <sup>[16]</sup>, i.e.,

$$\text{FOU}(\tilde{A}) = J_x, \quad x \in X. \quad (10)$$

The structure of a typical type-2 fuzzy logic system is shown in Fig. 3. It is similar to its type-1 counterpart. The major difference is that at least one of the fuzzy sets is type-2 and a type-reducer is needed to convert the type-2 fuzzy output sets into type-1 sets so that they can be processed by the defuzzify to give a crisp output<sup>[18]</sup>. General type-2 fuzzy logic systems (FLSs) are computationally intensive because type-reduction is very intensive<sup>[18, 29]</sup>. Therefore, we will use in this work the interval type-2 fuzzy logic systems for their simplicity and efficiency. We design the fuzzy controller by the use of two inputs namely tracking error and its derivative (derror), respectively. If we select three fuzzy sets for each input, the whole control space will be covered by 9 fuzzy rules. The fuzzy rules are of the form of

$$R^i : \text{if } x_1 \text{ is } \tilde{X}_1^i \text{ and } x_2 \text{ is } \tilde{X}_2^i, \\ \text{then } y^i = a_{i1}x_1 + a_{i2}x_2 + a_{i0}, \quad i = 1, 2, \dots, m \quad (11)$$

where  $\tilde{X}_j^i$  ( $j = 1, 2$ ) is an interval type-2 fuzzy set and the inputs of rule  $R^i$  is  $x = (x_1, x_2) \in U \in \mathbf{R}^2$ ,  $U$  is the universe of discourse.  $M$  is the number of rules, and in the  $i$ -th rule ( $R^i$ ),  $a_{i1}$  and  $a_{i2}$  are the gains in consequent part for  $i = 1, 2$ ,  $y^i$  is the crisp output. The proposed interval type-2 fuzzy controller is for the case when antecedents are interval type-2 fuzzy sets (A2) and consequents are crisp numbers (C0)<sup>[30]</sup>. The inference engine combines all the firing rules and gives a nonlinear mapping from the input interval type-2 fuzzy sets to the output interval type-2 fuzzy sets.

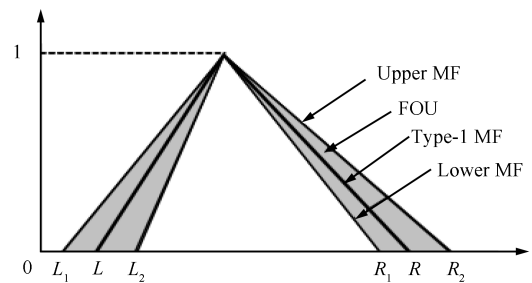


Fig. 2 Type-2 fuzzy logic membership function

The firing strength set of the  $i$ -th rule is<sup>[29]</sup>

$$F^i(x) = [\underline{f}^i(x), \bar{f}^i(x)] = [\underline{f}^i, \bar{f}^i] \quad (12)$$

where

$$\underline{f}^i(x) = \underline{\mu}_{\tilde{X}_1^i}(x_1) \times \underline{\mu}_{\tilde{X}_2^i}(x_2) \quad (13)$$

$$\bar{f}^i(x) = \bar{\mu}_{\tilde{X}_1^i}(x_1) \times \bar{\mu}_{\tilde{X}_2^i}(x_2). \quad (14)$$

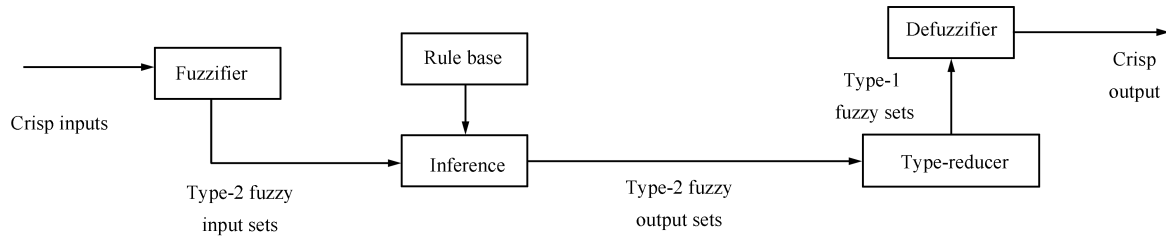


Fig. 3 Scheme of a type-2 fuzzy logic system

The terms  $\underline{\mu}_{\tilde{x}_j}(j = 1, 2)$  and  $\overline{\mu}_{\tilde{x}_j}(j = 1, 2)$  are the lower and upper membership grades of  $\mu_{\tilde{x}_j}$ , respectively.

The type-2 fuzzy inference engine produces an aggregated output type-2 fuzzy set. The type reduction block operates on this set to generate a centroid type-1 fuzzy set known as the “type-reduced set” of the aggregate type-2 fuzzy set. Several type-reduction methods have been suggested in the literature, such as the center-of-sums, the height, the modified height and the center-of-sets<sup>[18, 29]</sup>. The most commonly used one is the center-of-sets type-reducer due to its computational efficiency. That may be expressed as<sup>[18, 29]</sup>

$$Y_{\text{cos}}(x) = [y_l, y_r] = \bigcup_{f^i \in F^i(x)} \frac{\sum_{i=1}^M f^i y^i}{\sum_{i=1}^M f^i} \quad (15)$$

where  $Y_{\text{cos}}$  is the interval set determined by two end points  $y_l$  and  $y_r$ , and firing strengths  $f^i = [f^i, \overline{f}^i] \in F^i(x)$ .  $y_l$  and  $y_r$  can be expressed as

$$y_l = \frac{\sum_{i=1}^L \overline{f}^i y^i + \sum_{i=L+1}^M f^i y^i}{\sum_{i=1}^L \overline{f}^i + \sum_{i=L+1}^M f^i} \quad (16)$$

$$y_r = \frac{\sum_{i=1}^R \overline{f}^i y^i + \sum_{i=R+1}^M \overline{f}^i y^i}{\sum_{i=1}^R \overline{f}^i + \sum_{i=R+1}^M \overline{f}^i} \quad (17)$$

Two end points  $y_l$  and  $y_r$  can be computed efficiently using the Karnik-Mendel (KM) algorithms<sup>[29]</sup>. Since the type-reduced set is an interval type-1 set, the defuzzify output is<sup>[18]</sup>

$$y(x) = 0.5(y_l + y_r). \quad (18)$$

## 4 Particle swarm optimization (PSO)

The PSO algorithm is performed as follows. The unknown parameters are called the particles that form the population size denoted by  $n$ . Starting with a randomly initialization, the particles will move in a searching space to minimize an objective function. The parameters are estimated through minimizing the objective function. The fitness of each particle is evaluated according to the objective function for updating the best position of particle

and the best position among all particles as two goals in each step of computing. Each particle is directed to its previous best position and the previous best position among particles. Consequently, the particles tend to fly towards the better searching areas over the searching space. The velocity of  $i$ -th particle  $v_i$  will be calculated as<sup>[26, 27]</sup>

$$v_i(k+1) = \chi(v_i(k) + c_1 r_1 (pbest_i(k) - x_i(k)) + c_2 r_2 (gbest - x_i(k))) \quad (19)$$

where for the  $i$ -th particle in the  $k$ -th iteration,  $\chi_i$  is the position,  $pbest_i$  is the previous best position,  $gbest$  is the previous global best position of particles,  $c_1$  and  $c_2$  are the acceleration coefficients namely the cognitive and social scaling parameters,  $r_1$  and  $r_2$  are two random numbers in the range of  $[0, 1]$ ,  $\varphi = c_1 + c_2$ ,  $\varphi > 4$ , and  $\chi$  is constriction coefficient given by

$$\chi = \frac{2}{|4 - \varphi - \sqrt{\varphi^2 - 4\varphi}|} \quad (20)$$

Constriction coefficient is controlled by the convergence of the particle. As a result, it prevents explosion and ensuring convergence<sup>[26]</sup>.

A new position of the  $i$ -th particle is then calculated as

$$x_i(k+1) = x_i(k) + v_i(k+1). \quad (21)$$

The PSO algorithm performs repeatedly until the goal is achieved. The number of iterations denoted by  $k_{\text{max}}$  can be set to a specific value as a goal of optimization. The most crucial step in applying PSO is to choose the best cost functions which are used to evaluate the fitness of each particle. During tuning process with PSO, the mean of root of squared error (MRSE) is used.

The cost functions for the  $i$ -th particle are computed as

$$\text{MRSE} = E(k) = \frac{1}{N} \sum_{i=1}^N \sqrt{e_1^2(i) + e_2^2(i)} \quad (22)$$

where  $e_1(i)$  is the trajectory error of the  $i$ -th sample for the first joint,  $e_2(i)$  is the trajectory error of the  $i$ -th sample for the second joint,  $N$  is the number of samples, and  $k$  is the iteration number.

PSO algorithm initializes randomly. However, the speed of convergence is sensitive to the initialization. On the other hand, by using constriction coefficient, the PSO algorithm should find the optimum solution whatever the initial values

of the particles are<sup>[26]</sup>. In addition, the PSO runs off-line, thus computing time is not as important as in real time control. Since the control design uses the optimal solution provided by PSO, the PSO-based control strategy is not dependent on the initial values.

### 5 Control strategy

We adopt two-input-one-output FLC to introduce the design procedures of IT2FLC, i.e., we consider the error  $x_1$  and error rate  $x_2$  as inputs. Let  $x = \theta - \theta_d$ , where  $\theta$  is a vector of joint angles and  $\theta_d$  is a vector of the desired joint angles. The IT2FLC takes these two inputs and provides a control action  $u \in \mathbf{R}^2$ .

The membership functions of an FLC can be assumed as triangular functions, trapezoidal functions, Gaussian functions, etc<sup>[21]</sup>. In this study, Gaussian functions are used. Gaussian membership function has less parameters than the triangular one<sup>[21]</sup>, which is preferred for optimization. The membership functions adopted by both types of control

systems are shown in Figs. 4 and 5, respectively. Note that “N” stands for negative, “E” stands for 0, and “P” stands for positive. The rules of the fuzzy controllers are given in Section 6.

To assess the performance of both types of controllers, the proposed fuzzy controller is implemented in two different ways: the first is based on a type-1 fuzzy control scheme, while the second is based on a type-2 fuzzy control scheme. In the case of the implementation of the IT2FLC, we have the same characteristics as in T1FLC, but we used type-2 fuzzy sets as membership functions for the inputs as shown in Fig. 5.

All the universes of discourses are normalized and arranged in  $[-1 1]$ . Moreover, scaling factors external to the FLC used to give appropriate values to the variables. The role of input scaling factors becomes more important for using the Gaussian MFs for inputs. The input scaling factors are employed to take the input into the operating range covered by MFs, otherwise the controller will not respond to the input.

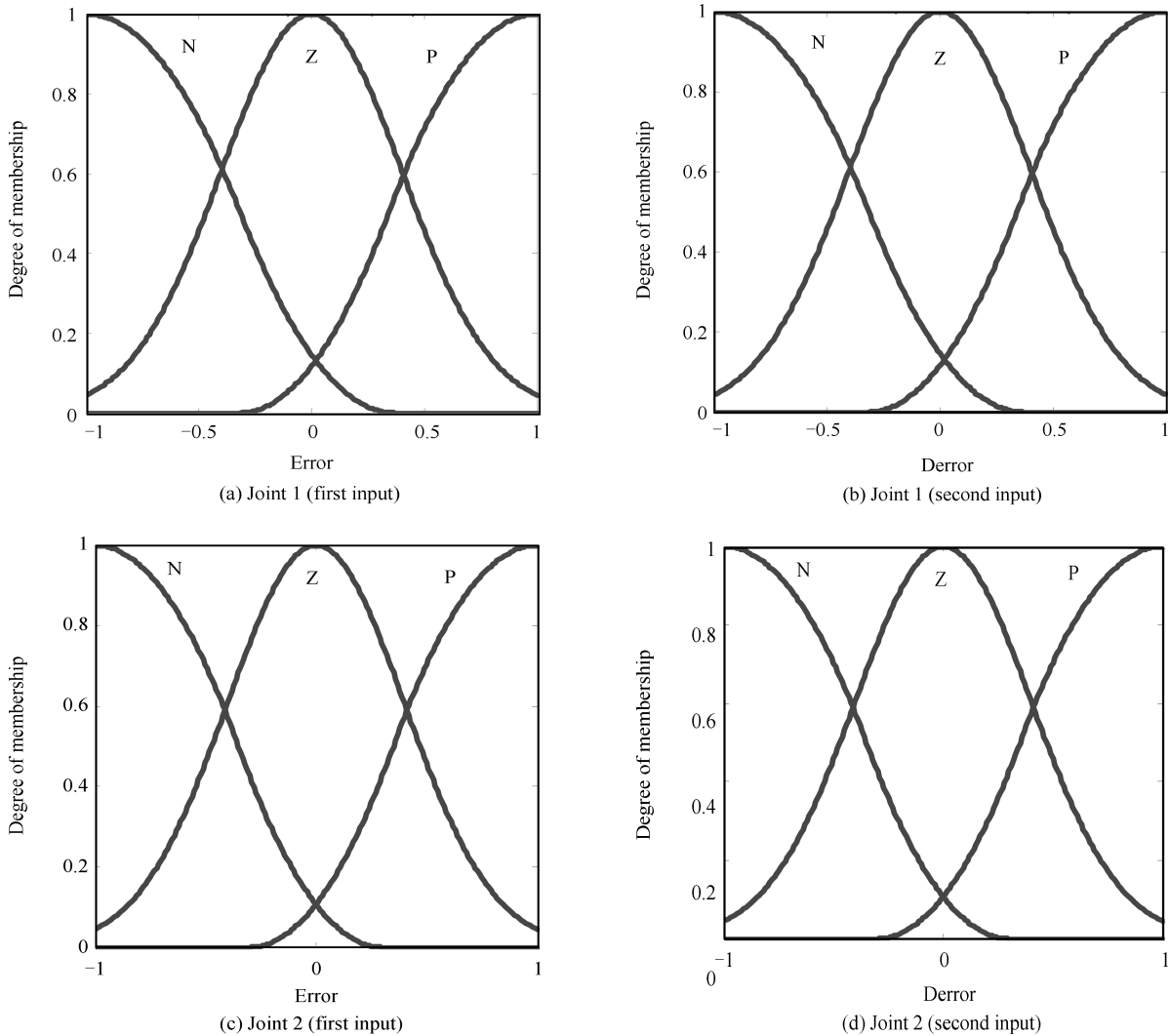


Fig. 4 Membership function of type-1 FLC for joint 1 (a,b) and joint 2 (c,d)

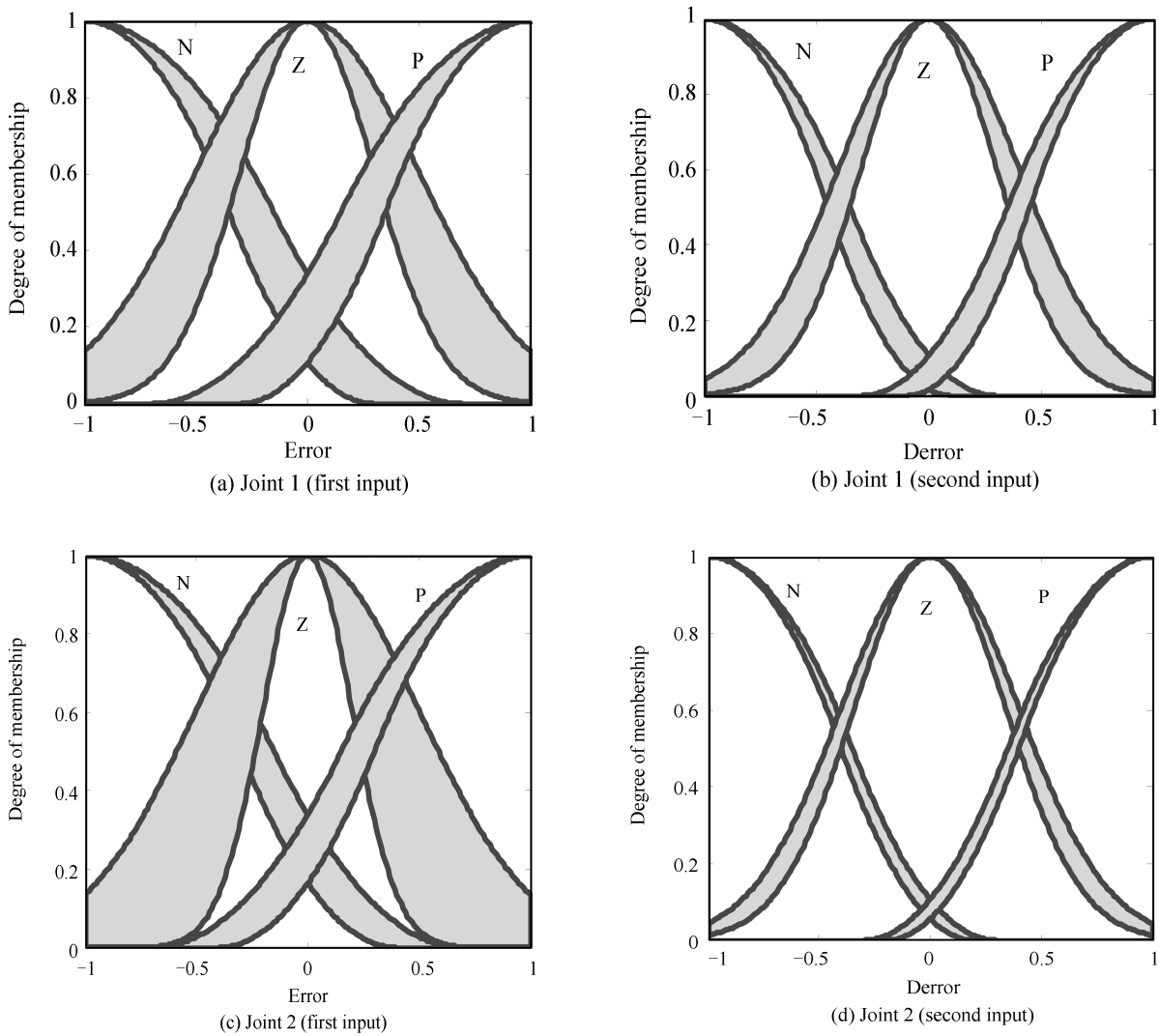


Fig. 5 Membership function of type-2 FLC for joint 1 (a,b) and joint 2 (c,d)

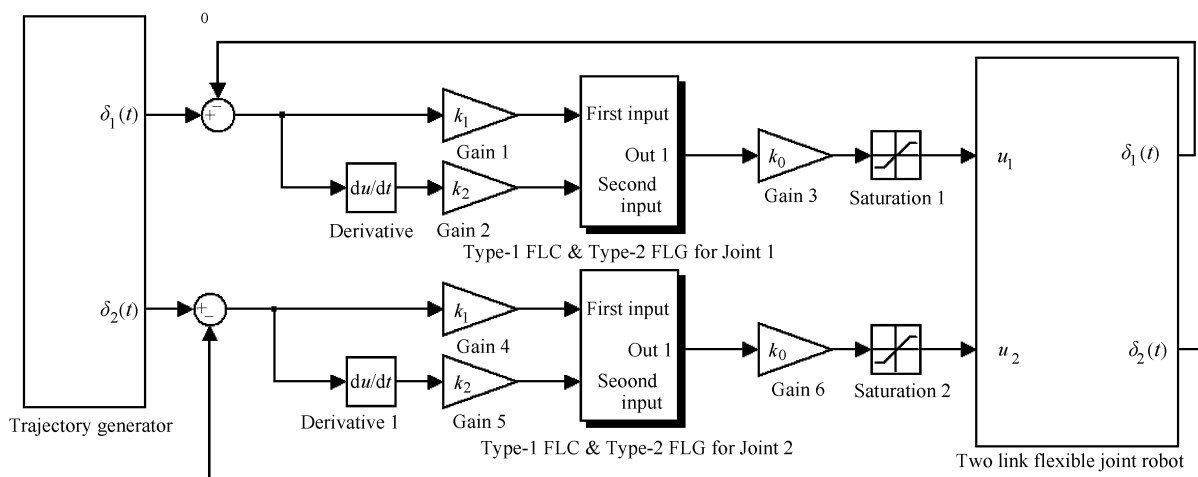


Fig. 6 Simulink model of proposed method

Input variables and output variable have scaling factors  $k_1, k_2, k_o$  and  $k'_1, k'_2$  and  $k'_o$  for joint 1 and joint 2, respectively. Simulink model of the proposed method is shown in Fig. 6.

A Gaussian MF with certain mean ( $m$ ) and uncertain variance ( $\delta$ ) can be completely defined by 3 parameters:  $m$  and  $(\delta_1, \delta_2)^{[29]}$ . Since a different IT2FLC is applied to each link and each IT2FLC has 6 input type-2 MFs and 3 different scaling factors, IT2FLC has a total of  $2 \times (6 \times 3 + 3) = 42$  parameters to be optimized. As the PSO will only tune the membership functions of both T1FLC and IT2FLC, the rules are fixed.

In the similar manner, since a different T1FLC is applied to each link, two parameters are sufficient to determine a Gaussian type-1 MF, and 3 different scaling factors, the PSO has to optimize a total of  $2 \times (6 \times 2 + 3) = 30$  parameters. As T1FLC and IT2FLC have the same number of MFs and rules, comparing their performance may provide insights into the contributions made by the FOU.

Using the output scaling factor  $k_0$ , the fuzzy controller is formed as

$$u = k_0 y(x). \quad (23)$$

Each joint is controlled separately using fuzzy controller (23). Thus,

$$u = [u_1 \quad u_2]^T \quad (24)$$

where  $u_1$  and  $u_2$  are the fuzzy controllers for joint 1 and 2, respectively as expressed in (23).

## 6 Stability analysis of the control system

In other words,  $y_l$  in (16) can be rewritten as

$$y_l = \sum_{i=1}^L \bar{q}_l^i (a_{i1}x_1 + a_{i2}x_2 + a_{i0}) + \sum_{i=L+1}^M \underline{q}_l^i (a_{i1}x_1 + a_{i2}x_2 + a_{i0}) \quad (25)$$

where  $\bar{q}_l^i = \frac{\bar{f}^i}{D_l}$  and  $\underline{q}_l^i = \frac{f^i}{D_l}$ . In the meantime, we have  $D_l = \sum_{i=1}^L \bar{f}^i + \sum_{i=L+1}^M f^i$ .

In the similar manner,  $y_r$  in (17) can be rewritten as

$$y_r = \sum_{i=1}^R \underline{q}_r^i (a_{i1}x_1 + a_{i2}x_2 + a_{i0}) + \sum_{i=R+1}^M \bar{q}_r^i (a_{i1}x_1 + a_{i2}x_2 + a_{i0}) \quad (26)$$

where  $\bar{q}_r^i = \frac{\bar{f}^i}{D_r}$  and  $\underline{q}_r^i = \frac{f^i}{D_r}$ . In the meantime, we have  $D_r = \sum_{i=1}^R \underline{f}^i + \sum_{i=R+1}^M \bar{f}^i$ .

From (18), after some manipulation, one can obtain

$$y(x) = C_1(x)x_1 + C_2(x)x_2 + C_0(x) \quad (27)$$

where

$$C_1(x) = 0.5 \times \left( \sum_{i=1}^L \bar{q}_l^i a_{i1} + \sum_{i=L+1}^L \underline{q}_l^i a_{i1} + \sum_{i=1}^R \underline{q}_r^i a_{i1} + \sum_{i=R+1}^M \bar{q}_r^i a_{i1} \right) \quad (28)$$

$$C_2(x) = 0.5 \times \left( \sum_{i=1}^L \bar{q}_l^i a_{i2} + \sum_{i=L+1}^L \underline{q}_l^i a_{i2} + \sum_{i=1}^R \underline{q}_r^i a_{i2} + \sum_{i=R+1}^M \bar{q}_r^i a_{i2} \right) \quad (29)$$

$$C_0(x) = 0.5 \times \left( \sum_{i=1}^L \bar{q}_l^i a_{i0} + \sum_{i=L+1}^L \underline{q}_l^i a_{i0} + \sum_{i=1}^R \underline{q}_r^i a_{i0} + \sum_{i=R+1}^M \bar{q}_r^i a_{i0} \right). \quad (30)$$

The obtained analytical structure of the fuzzy controller improves our study to develop the analysis and design. To normalize the controller, we use the input scaling factors  $k_1 > 0$  and  $k_2 > 0$ , and the output scaling factor  $k_0 > 0$ . The input vector is formed as

$$x = [k_1 z_1 \quad k_2 z_2]^T \quad (31)$$

where for the  $i$ -th joint,  $z_1$  and  $z_2$  are defined as

$$z_1 = \theta_{di} - \theta_i \quad (32)$$

$$z_2 = \dot{\theta}_{di} - \dot{\theta}_i \quad (33)$$

where  $\theta_{di}$  and  $\theta_i$  are the desired and actual joint position, respectively. From (32) and (33), we have  $\dot{z}_1 = z_2$ . Using  $x_1 = k_1 z_1$  and  $x_2 = k_2 z_2$ , we obtain

$$\dot{x}_1 = \alpha x_2 \quad (34)$$

where  $\alpha = \frac{k_1}{k_2} > 0$ .

Fuzzy controller by the use of scaling factors is formed as

$$u(x) = k_0 (C_1(x)x_1 + C_2(x)x_2 + C_0(x)). \quad (35)$$

This general structure shows a nonlinear variable gain controller that finds many applications in control. The nonlinear gain  $C_i(x)$  covers the nonlinearity of the controller by parameters in hand. The control purposes are simply described by linguistic rules in fuzzy controller transformed to a nonlinear function as stated by (35).

Substituting (35) into (7) forms the closed loop system as

$$(C_1(x)x_1 + C_2(x)x_2 + C_0(x))k_0 = RI_a + LI_a + K_b \dot{\theta}_m. \quad (36)$$

Assume that the motor voltage  $u$  expressed by (7) is limited such that

$$|RI_a + LI_a + K_b \dot{\theta}_m| \leq u_{\max} \quad (37)$$

where  $u_{\max} > 0$  is a maximum permitted voltage for the motor. This assumption is a technical regard to protect the

motor against over-voltages. The complexity of design and analysis has been changed to simplicity by using the model of motor in place of model of manipulator. Here, we should know only the upper limits of the motor voltages ( $u_{\max}$ ) as inputs of robotic system. Because electrical motors drive the electrical manipulator, the motor voltages are the system inputs. The desired trajectory should be planned considering the maximum permitted voltages for motors. This means that each motor is so strong that it can track the desired trajectory under the permitted voltage. Moreover, the desired trajectory should be smooth that its derivatives up to the required order are available and limited. To find a control law for the convergence of error, a positive definite function is proposed as

$$V = \int_0^{x_1} C_2(x)x_1 dx_1 \tag{38}$$

where  $V$  is a positive definite function of  $x_1$  if  $C_2(x)$  is positive. To satisfy  $C_2(x) > 0$ , it is sufficient that  $a_{i2} \geq 0$ .

**Proof.** Assume that  $0 < C_2 < C_2(x)$ , where  $C_2$  is a positive constant. Thus,

$$C_2 \int_0^{x_1} x_1 dx_1 < \int_0^{x_1} C_2(x)x_1 dx_1. \tag{39}$$

We have  $\int_0^{x_1} C_2 x_1 dx_1 = 0.5C_2 x_1^2$ . Hence,  $0.5C_2 x_1^2 < \int_0^{x_1} C_2 x_1 dx_1$ . Thus, (39) implies that  $V > 0$  for  $x_1 \neq 0$ . Since  $\int_0^{x_1} C_2 x_1 dx_1 = 0$  and  $C_2(x)x_1$  is limited,  $V = 0$  if  $x_1 = 0$ . Thus,  $V$  is a positive definite function of  $x_1$ .  $\square$

The time derivative of  $V$  is calculated as

$$\dot{V} = C_2(x)x_1\dot{x}_1 = \alpha C_2(x)x_1x_2. \tag{40}$$

From (36), we can write

$$C_2(x)x_2 = -C_1(x)x_1 - C_0(x) + \frac{RI_a + LI_a + K_b\dot{\theta}_m}{k_0}. \tag{41}$$

Substituting (41) into (40) yields

$$\begin{aligned} \dot{V} = & -\alpha C_1(x)x_1^2 - \alpha C_0(x)x_1 + \\ & \alpha x_1 \times \frac{(RI_a + LI_a + K_b\dot{\theta}_m)}{k_0} \end{aligned} \tag{42}$$

Since  $-\alpha C_1(x)x_1^2 \leq 0$  for  $C_1(x) \geq 0$ , to satisfy  $\dot{V} \leq 0$  for stability, it is required that

$$x_1(RI_a + LI_a + K_b\dot{\theta}_m) \leq k_0 C_0(x)x_1. \tag{43}$$

As we known,

$$\begin{aligned} x_1(RI_a + LI_a + K_b\dot{\theta}_m) & \leq \\ |x_1|(RI_a + LI_a + K_b\dot{\theta}_m) & \leq |x_1|u_{\max}. \end{aligned} \tag{44}$$

To satisfy (43), it is sufficient that

$$|x_1|u_{\max} \leq k_0 C_0(x)x_1. \tag{45}$$

Since  $k_0 > 0$ , to guarantee the stability,  $x_1 C_0(x)$  must be positive. This means that  $C_0(x)$  must be designed with the same sign as  $x_1$ . This condition is simply satisfied if  $a_{i0}$  is selected with the same sign as  $x_1$ .

From (45) and  $x_1 C_0(x) > 0$ , we obtain

$$\frac{u_{\max}}{|C_0(x)|} \leq k_0. \tag{46}$$

From (30), we can obtain

$$c_{0,\min} \leq C_0(x) \leq c_{0,\max} \tag{47}$$

where  $c_{0,\min}$  and  $c_{0,\max}$  are two constants. To select a constant value, we should select a value for  $k_0$  that satisfies (46) in all the cases. The maximum permitted value for  $k_0$  is then selected as

$$\frac{u_{\max}}{c_{0,\max}} = k_0. \tag{48}$$

Therefore, stability is guaranteed under assumptions  $C_1(x) > 0, C_2(x) > 0, x_1 C_0(x) > 0$  and  $\frac{u_{\max}}{c_{0,\max}} = k_0$ . In the  $i$ -th rule,  $a_{i0}$  is selected with the same sign as  $x_1$  to satisfy  $x_1 C_0(x) > 0$ . We can select  $a_{i1} \geq 0$  and  $a_{i2} \geq 0$  in all rules but to satisfy  $C_2(x) > 0$  and  $x_1 C_0(x) > 0$ . There is at least one rule that  $a_{i1} > 0$  and  $a_{i2} > 0$ .

Fuzzy rules for  $i = 1, \dots, 9$  are designed where the following cases occur:

Case 1. Assume that  $x_1 x_2 < 0$  result in asymptotic stability by  $\dot{V} < 0$  in (40). Thus,  $u$  should be small.

Case 2. Assume that  $x_1 = 0$  or  $x_2 = 0$  result in Lyapunov stability by  $\dot{V} = 0$  in (40). Thus,  $u$  should be medium.

Case 3. Assume that  $x_1 x_2 > 0$  result in instability by  $\dot{V} > 0$  in (40). Thus,  $u$  should be very high.

The fuzzy rules for the first and second controllers are given as follows:

Rule 1: If  $x_1$  is  $P$  and  $x_2$  is  $P$ , then  $y = 1$ .

Rule 2: If  $x_1$  is  $P$  and  $x_2$  is  $Z$ , then  $y = 0.75$ .

Rule 3: If  $x_1$  is  $P$  and  $x_2$  is  $N$ , then  $y = 0.25$ .

Rule 4: If  $x_1$  is  $Z$  and  $x_2$  is  $P$ , then  $y = 0.5$ .

Rule 5: If  $x_1$  is  $Z$  and  $x_2$  is  $Z$ , then  $y = 150x_1 + 10x_2$ .

Rule 6: If  $x_1$  is  $Z$  and  $x_2$  is  $N$ , then  $y = -0.5$ .

Rule 7: If  $x_1$  is  $N$  and  $x_2$  is  $P$ , then  $y = -0.25$ .

Rule 8: If  $x_1$  is  $N$  and  $x_2$  is  $Z$ , then  $y = -0.75$ .

Rule 9: If  $x_1$  is  $N$  and  $x_2$  is  $N$ , then  $y = -1$ .

Therefore, using the above analysis,  $x_1$  and  $x_2$  are bounded. Then, one can imply the boundedness of  $u$  because of boundedness of  $x_1$  and  $x_2$ .

**Proof.** From (28) to (30),  $C_1(x)$ ,  $C_2(x)$  and  $C_0(x)$  are bounded as

$$|C_1(x)| \leq 0.5\alpha_1|x_1| \tag{49}$$

$$|C_2(x)| \leq 0.5\alpha_2|x_2| \tag{50}$$

$$|C_0(x)| \leq 0.5\alpha_0 \tag{51}$$



where

$$\left| \sum_{i=1}^L \bar{q}_l^i a_{i1} + \sum_{i=L+1}^M \underline{q}_l^i a_{i1} + \sum_{i=1}^R \underline{q}_r^i a_{i1} + \sum_{i=R+1}^M \bar{q}_r^i a_{i1} \right| \leq \alpha_1 \quad (52)$$

$$\left| \sum_{i=1}^L \bar{q}_l^i a_{i2} + \sum_{i=L+1}^M \underline{q}_l^i a_{i2} + \sum_{i=1}^R \underline{q}_r^i a_{i2} + \sum_{i=R+1}^M \bar{q}_r^i a_{i2} \right| \leq \alpha_2 \quad (53)$$

$$\left| \sum_{i=1}^L \bar{q}_l^i a_{i0} + \sum_{i=L+1}^M \underline{q}_l^i a_{i0} + \sum_{i=1}^R \underline{q}_r^i a_{i0} + \sum_{i=R+1}^M \bar{q}_r^i a_{i0} \right| \leq \alpha_0 \quad (54)$$

and  $\alpha_1, \alpha_2, \alpha_0$  are constant.

Considering (13) and (14), we have

$$\underline{f}^i(x) \leq 1 \quad (55)$$

$$\bar{f}^i(x) \leq 1. \quad (56)$$

Thus, one can imply that  $\bar{q}_l^i, \underline{q}_l^i, \bar{q}_r^i$  and  $\underline{q}_r^i$  are bounded. The coefficient  $a_{i1}$  is a constant parameter. As a result, inequality (52) is verified.

Similarly, inequality (53) and (54) are proven.  $\square$

Therefore,  $u$  is bounded using (35) as

$$|u| \leq k_0(\alpha_1|x_1| + \alpha_2|x_2| + \alpha_0). \quad (57)$$

According to the proof given by [31], since the input  $u$  is a bounded variable,  $I_a$  is bounded.

Since the desired joint angle  $\theta_d$  and its time derivative  $\dot{\theta}_d$  are bounded, the bounded variables  $x_1$  and  $x_2$  imply that  $\theta = \theta_d - x_1$  and  $\dot{\theta} = \dot{\theta}_d - x_2$  are bounded.

Since  $I_a$  is bounded, (4) implies that  $\tau$  is a bounded. From (2), we have

$$J\ddot{\theta}_m + B\dot{\theta}_m + r^2K\theta_m = \tau + rK\theta. \quad (58)$$

System (58) is a second order linear system with positive gains  $J, B, r^2K$ , and a limited input  $\tau + rK\theta$ . This system is stable based on the Routh-Hurwitz criterion and implies that  $\theta_m, \dot{\theta}_m$  and  $\ddot{\theta}_m$  are bounded.

Since all the states associated with each joint, i.e.,  $\theta, \dot{\theta}, \theta_m, \dot{\theta}_m$  and  $I_a$ , are bounded, vectors  $\theta, \dot{\theta}, \theta_m, \dot{\theta}_m$  and  $I_a$  are bounded. As a conclusion, based on the stability analysis, all the required signals are bounded.

## 7 Simulation results

In this section, to compare the performance of T1FLC and IT2FLC, the performance of position and tracking control of an electrical flexible-joint articulated robot manipulator are presented. Fig. 6 shows the control system used for obtaining the simulation results. The parameters of the manipulator and motors are given in Tables 1 and 2, respectively. Moreover, the voltage of motors is limited to  $\pm 40$  V to protect motor from over-voltage. The desired joint trajectory for the joints is shown in Fig. 7. The desired

trajectory should be sufficiently smooth such that all its derivatives up to the required order are bounded.

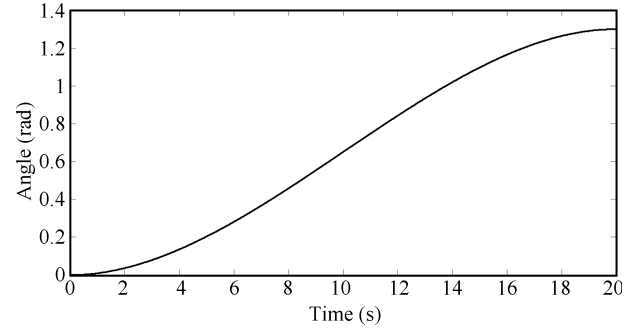


Fig. 7 Desired joint trajectory

Table 1 The dynamical parameters of robot

Links	$L$ (m)	$L_c$ (m)	$m$	$I$ (kg · m <sup>2</sup> )	$b$	$c$
1	1	0.5	1	0.12	1	5
2	1	0.5	2	0.25	1	5

Table 2 The motor parameters

Motors	$u$	$R$	$k_b$	$L$	$I$	$B$	$r$	$k$
1.2	40	1.6	0.26	0.001	0.0002	0.001	0.02	500

**Simulation 1.** The control system is simulated for set point control, where a desired value of rad is given to the joints. Set point application is used for point-to-point motion control of robot manipulators. In industry, the set point control is also a dominant approach in the process control. Scaling factors are given in Table 3. To have satisfactory performance, the gains of input and output scaling factors are selected by trial and error method.

Table 3 Scaling factors of type-1 and type-2 FLC

Scaling factor controller	$k_1$	$k_2$	$k_0$	$k'_1$	$k'_2$	$k'_0$
Type-1 FLC	0.003	35	100	0.0027	25	93
Type-2 FLC	0.002	36	105	0.002	28	111

The control system responses using T1FLC and IT2FLC are shown in Fig. 8. As shown in Fig. 8, the performance of the IT2FLC is better than T1FLC to some extent. However, the transient responses of both type of controllers are not good. The responses have overshoot with zero steady state error. An optimization algorithm, such as particle swarm optimization algorithm, can be applied to find optimum values for control design parameters to achieve the desired performance.

**Simulation 2.** In this simulation, the performance of controllers in tracking a desired trajectory is compared.

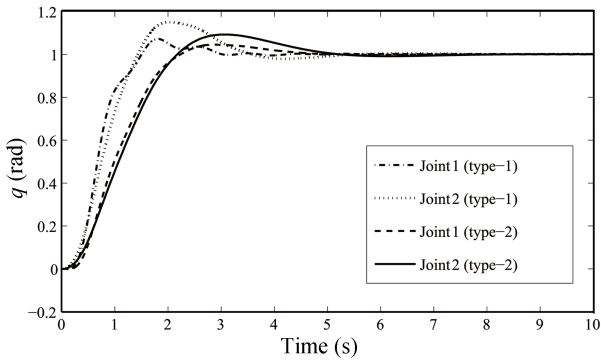


Fig. 8 Control system responses using type-1 and type-2 FLC

The desired joint trajectory for the joints is shown in Fig. 7. Other conditions are the same as Simulation 1. The control system performs well as shown in Fig. 9, while the maximum values of errors for joints 1–2 are 0.022 rad, 0.02 rad, and 0.0224 rad, 0.022 rad, using T1FLC and IT2FLC, respectively. The motor voltages of both joints are shown in Fig. 10. All of the voltages of motors lie under the valid limited values without chattering problem. It is seen that there is no change in voltage polarity, which offers good reasons to tolerate the high oscillations. In this case, the control efforts perform well. However, the joint tracking errors are not very good.

**Simulation 3.** As mentioned earlier, the control design parameters were obtained by trial and error method. However, we can use the proposed PSO algorithm to obtain better performance by finding the optimal control design parameters. In this simulation, the parameters of PSO algo-

rithm presented in Section 4 are set to  $c_1 = 2.05, c_2 = 2.05$  and  $\varphi = 0.72$ . The number of particles and iterations are set as 150 and 100, respectively. As a result of applying the PSO algorithm, the scaling factors of the optimized T1FLC and IT2FLC are reported in Table 4. Moreover, the MFs of the optimized IT2FLC and T1FLC are shown in Figs. 11 and 12, respectively.

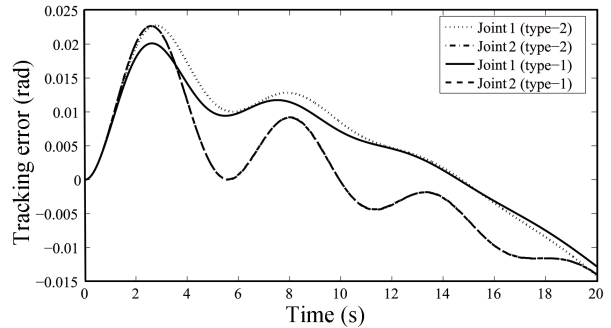


Fig. 9 Tracking performance of type-1 & type-2 FLC

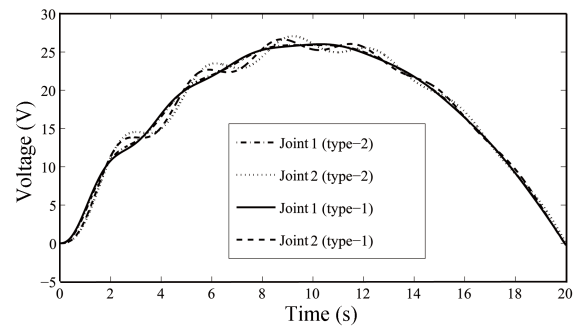
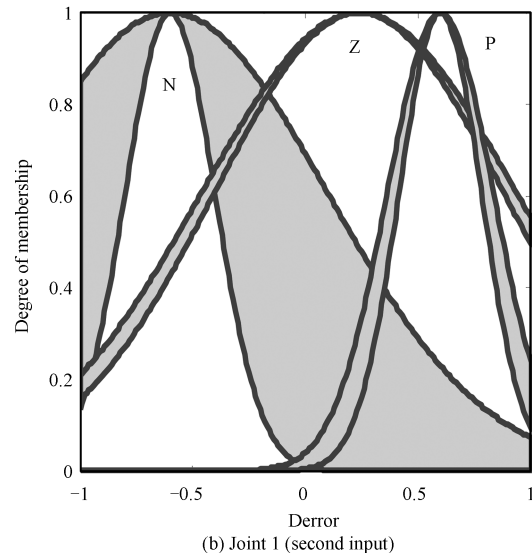
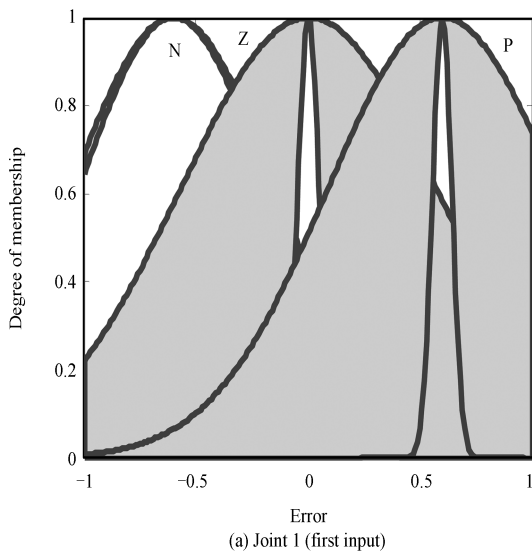


Fig. 10 Voltage of motors



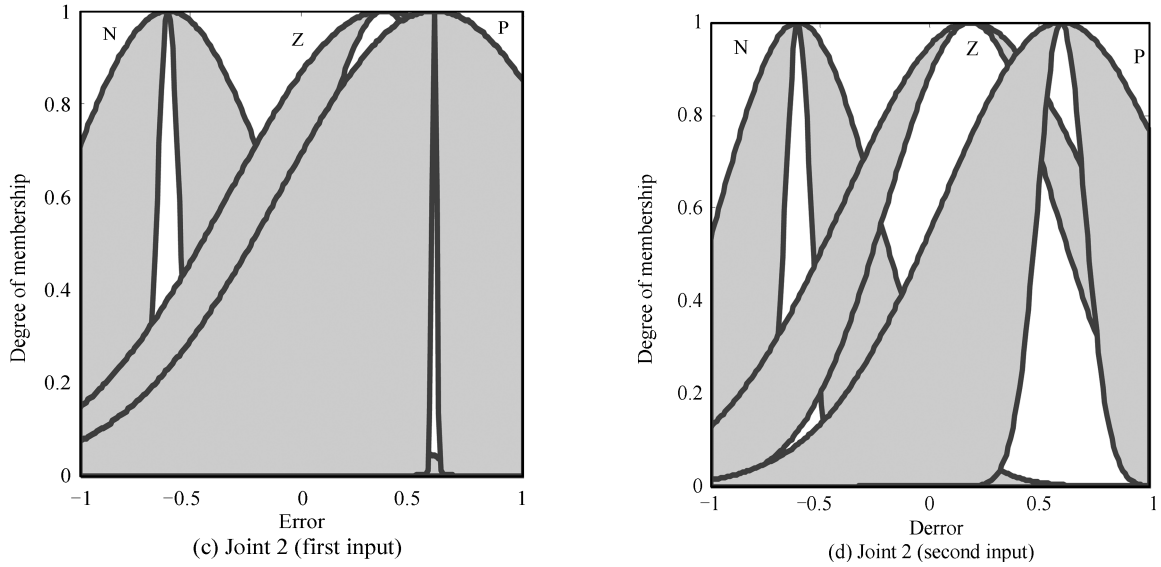


Fig. 11 Membership functions of the optimized type-2 FLC for joint 1 (a,b) and joint 2 (c,d)

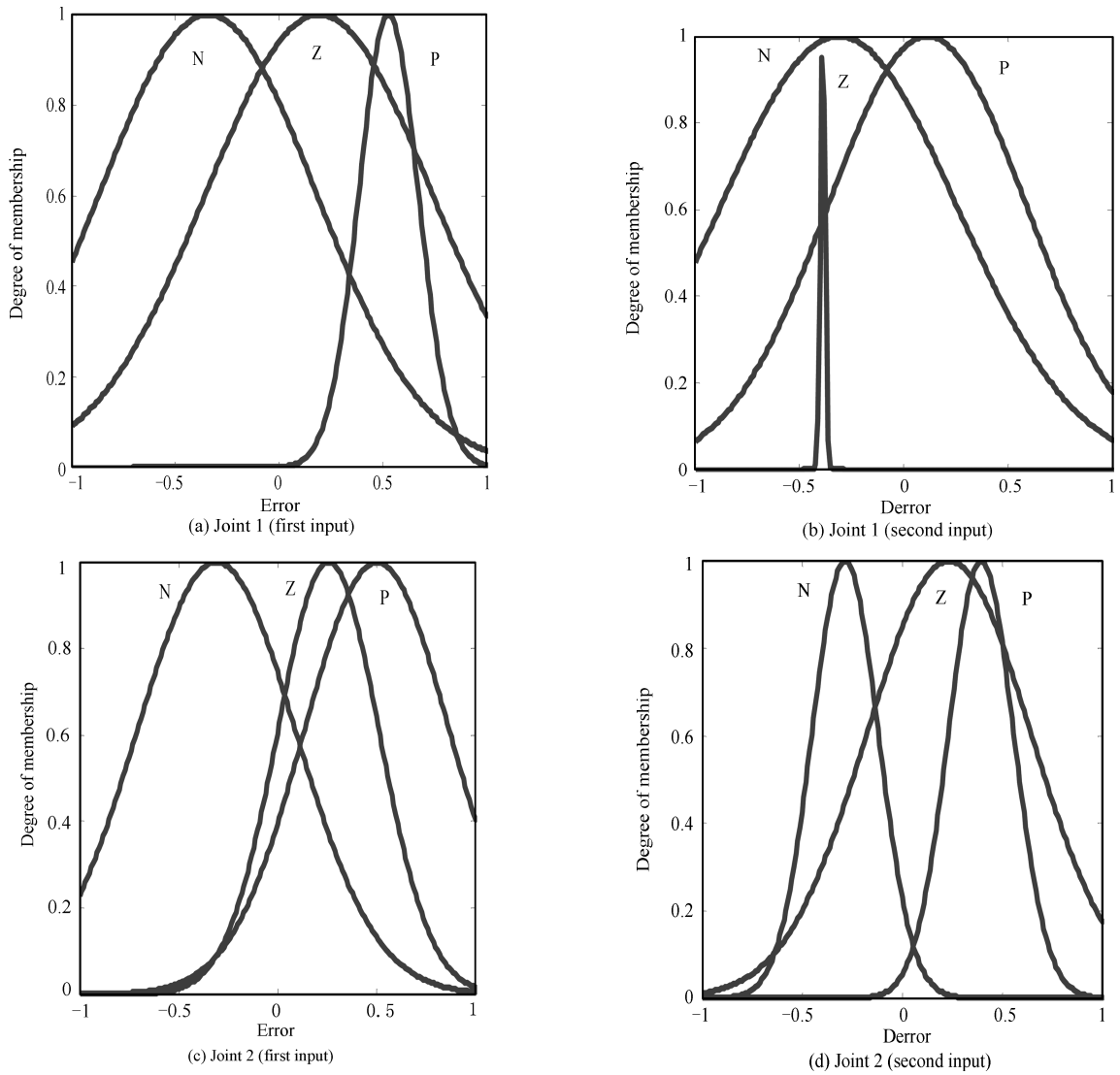


Fig. 12 Membership functions of the optimized type-1 FLC for joint 1 (a,b) and joint 2 (c,d)

Table 4 Scaling factors of the optimized type-1 and type-2 FLC

Optimal scaling factors controller	$k_1$	$k_2$	$k_0$	$k'_1$	$k'_2$	$k'_0$
Type-1 FLC	0.0021	35.04	103.5	0.0017	33.86	99.67
Type-2 FLC	0.0022	36.67	129.67	0.0018	37.08	120.89

The speed of convergence versus iteration is shown in Fig. 13. It indicates that the fitness values have converged. Another observation is that the additional mathematical dimension provided by the FOU enables IT2FLC to achieve a lower MRSE than T1FLC. Set point performance and tracking errors are shown in Figs. 14 and 15, respectively. The control system behaves well in these cases, as well. As a result of applying optimized T1FLC, the maximum value of tracking error for joint 1 and 2 in the simulations is reached to the value of  $7 \times 10^{-3}$  and  $8 \times 10^{-3}$ , respectively. As a result of applying optimized IT2FLC, the maximum value of tracking error for joint 1 and 2 in the simulations is reached to the value of  $6 \times 10^{-3}$  and  $7.98 \times 10^{-3}$ , respectively. It is seen that the results are much better than simulation 1. To assess the performance of control system better, another desired trajectory is considered as

$$\theta_d = \sin(t), \quad \theta(0) = 1, \quad 0 \leq t \leq 10. \quad (59)$$

The performance of control system is shown in Fig. 16, where the joint tracking error is reduced as well.

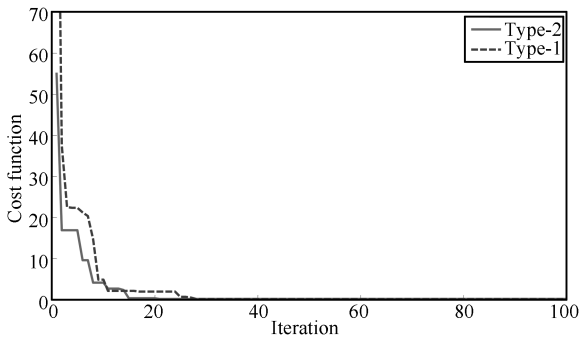


Fig. 13 The cost function of PSO

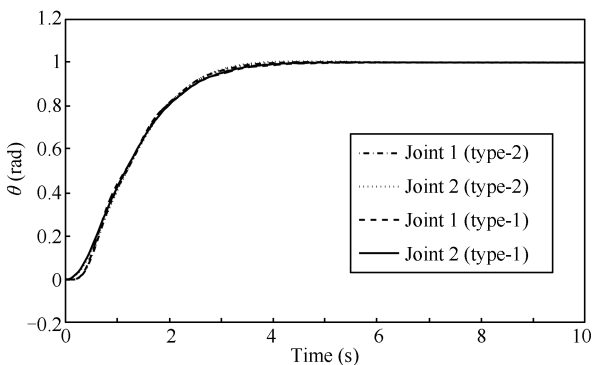


Fig. 14 Set point performance

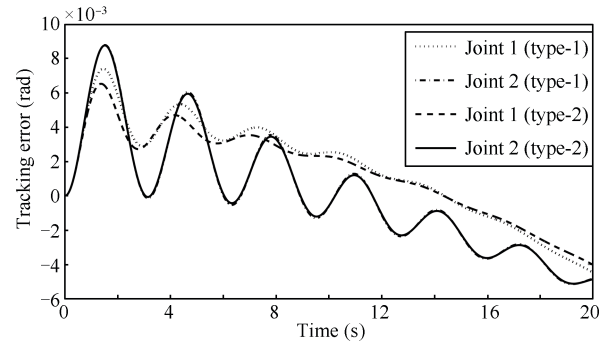


Fig. 15 Tracking performance

**Simulation 4.** In this case, for the robustness evaluation of the controllers, external disturbances are added to the robot system. The disturbance is inserted to the input of each motor as a periodic pulse function with a period of 2s, having a different value of amplitude, with time delay of 0.7s, and pulse-width 30% of the period<sup>[32]</sup>. This form of disturbance is general. But it includes jumps to cover the complex cases. All the values of the cost functions for different amplitude of disturbance in the cases of T1FLC and IT2FLC are given in Table 5.

Table 5 Comparison between type-1 and type-2 FLC for different amplitudes of disturbance

Amplitude	Type-1 FLC (MRSE)	Type-2 FLC (MRSE)
A=0	0.1892	0.1860
A=1	0.1958	0.1879
A=2	0.2044	0.1910
A=4	0.2337	0.2022

As seen from Table 5, the performance of the T1FLC is similar to IT2FLC to some extent as long as the amplitude of disturbance is low or there is no disturbance. However, when the amplitude of disturbance in the Table 5 is increased, it is seen that the performance of T1FLC degrades significantly. Hence, the performance of the T1FLC will be unacceptable under this disturbance. Thus, the T1FLC cannot be used in such noisy environment. On the other hand, IT2FLC can handle the external disturbances to give a very good performance effectively. As a result, the IT2FLC is able to handle the uncertainty and outperform the type-1 controller.

**Simulation 5.** In this case, the ability of the T1FLC and IT2FLC to handle unmodelled dynamics is investigated. To this end, transport delay is deliberately introduced into the feedback loop<sup>[20]</sup>. First, a transport delay equalling to 0.2s is artificially added to the nominal system. The step responses are shown in Fig. 17. When a the transport delay equalling to 0.3s is added to the system, the corresponding

step responses are shown in Fig. 18. As shown in Figs. 17 and 18, large oscillations are obtained when T1FLC is used to control the plant. Of the two controllers, IT2FLC provided the best performance as its step responses have the smallest overshoot and are least oscillatory. The simulation results again confirm that the IT2FLC is able to handle the uncertainty and unmodelled dynamics, and outperform the T1FLC controller.

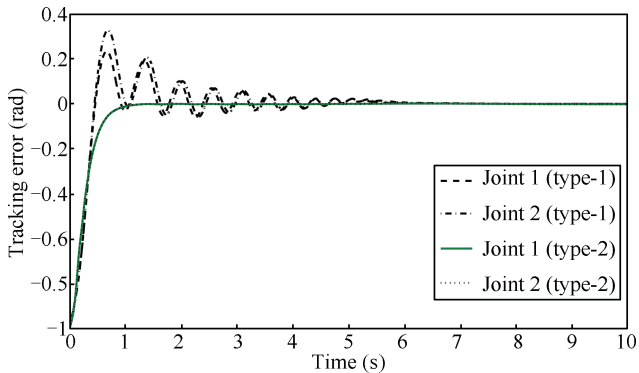


Fig. 16 Tracking performance

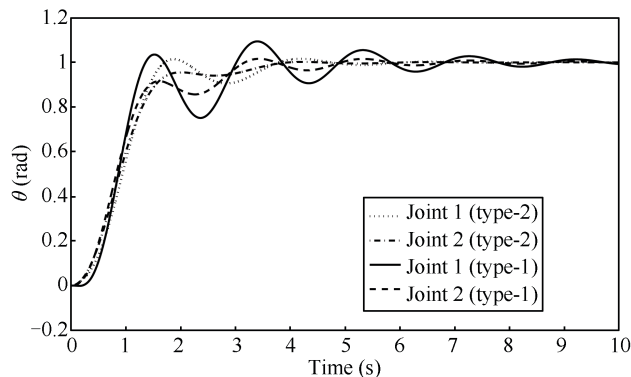


Fig. 17 Set point performance under unmodelled dynamics with time delay 0.2s

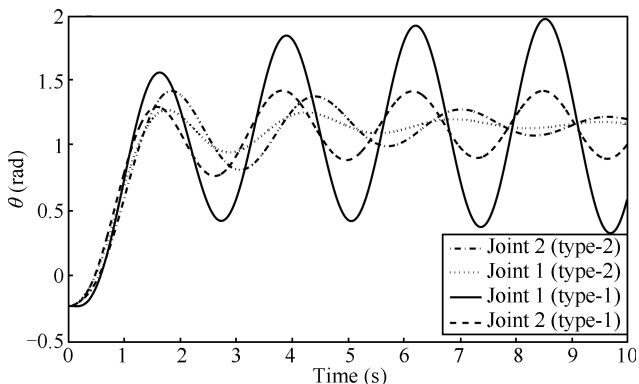


Fig. 18 Set point performance under unmodelled dynamics with time delay 0.3s

As a result, the main advantage of the IT2FLC is its ability to eliminate persistent oscillations, especially when unmodelled dynamics is introduced.

## 8 Conclusions

In this paper, interval type-2 fuzzy logic controller was used to control a two-joint articulated flexible-joint robot driven by permanent magnet direct current motors using voltage control strategy. Stability analysis was presented and the performance of the IT2FLC was compared with that of T1FLC. In addition, an optimal IT2FLC for flexible-joint robot was introduced using particle swarm optimization. In fact, parameters of the primary membership functions of IT2FLC were optimized to improve the performance and increase the accuracy of IT2FLC. We observed using performance criteria such as MRSE, when the amplitude of external disturbance is low, the performance of both T1FLC and IT2FLC is somehow identical. However, it is known that T1FLC can handle the nonlinearities and uncertainties up to some extent. Therefore, by increasing amplitude of external disturbance and considering unmodelled dynamics, the results demonstrate that IT2FLC can outperform T1FLC. Thus, the IT2FLC is more appealing than its type-1 counterpart with regards to accuracy and interpretability. The main advantage of the IT2FLC appears to be its ability to eliminate the persistent oscillations, especially when unmodelled dynamics is introduced. This ability to handle modeling error is particularly useful when fuzzy logic controllers are tuned off-line using PSO.

## Acknowledgments

The authors would like to thank Dr. Dong Wu for his valuable advices.

## References

- [1] B. Brogliato, R. Ortega, R. Lozano. Global tracking controllers for flexible-joint manipulators: A comparative study. *Automatica*, vol. 31, no. 7, pp. 941–956, 1995.
- [2] M. O. Tokhi, A. K. M. Azad. *Flexible Robot Manipulators: Modeling, Simulation and Control*, London: The Institution of Engineering and Technology, 2008.
- [3] J. O'Reilly, P. Kokotovic, H. K. Khalil. *Singular Perturbation Methods in Control: Analysis and Design*, New York: Academic Press, 1986.
- [4] A. De Luca, A. Isidori, F. Nicolo. Control of robot arm with elastic joints via nonlinear dynamic feedback. In *Proceedings of the 24th IEEE Conference on Decision Control*, IEEE, Ft. Lauderdale, FL, USA, pp. 1671–1679, 1985.
- [5] M. W. Spong. Adaptive control of flexible joint manipulators: Comments on two papers. *Automatica*, vol. 31, no. 4, pp. 585–590, 1995.
- [6] G. A. Wilson. Robust tracking of elastic joint manipulators using sliding mode control. *Transactions of the Institute of Measurement and Control*, vol. 16, no. 2, pp. 99–107, 1994.
- [7] M. W. Spong. Modeling and control of elastic joint robots. *Journal of Dynamic Systems, Measurement, and Control*, vol. 109, pp. 310–319, 1987.
- [8] L. C. Lin, C. C. Chen. Rigid model-based fuzzy control of flexible-joint manipulators. *Journal of Intelligent and Robotic Systems*, vol. 13, no. 2, pp. 107–126, 1995.

- [9] V. Zeman, R. V. Patel, K. Khorasani. Control of a flexible-joint robot using neural networks. *IEEE Transactions on Control Systems Technology*, vol. 5, no. 4, pp. 453–462, 1997.
- [10] M. M. Fateh. Robust control of flexible-joint robots using voltage control strategy. *Nonlinear Dynamics*, vol. 67, no. 2, pp. 1525–1537, 2012.
- [11] M. M. Fateh. Robust voltage control of electrical manipulators in task-space. *International Journal of Innovative Computing Information and Control*, vol. 6, no. 6, pp. 2691–2700, 2010.
- [12] M. M. Fateh. On the voltage-based control of robot manipulators. *International Journal of Control, Automation, and Systems*, vol. 6, no. 5, pp. 702–712, 2008.
- [13] M. M. Fateh. Nonlinear control of electrical flexible-joint robots. *Nonlinear Dynamics*, vol. 67, no. 4, pp. 2549–2559, 2012.
- [14] L. Cheng, Z. G. Hou, M. Tan. Adaptive neural network tracking control for manipulators with uncertain kinematics, dynamics and actuator model. *Automatica*, vol. 45, no. 10, pp. 2312–2318, 2009.
- [15] F. Karray, W. Gueaieb, S. Al-Sharhan. The hierarchical expert tuning of PID controllers using tools of soft computing. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 32, no. 1, pp. 77–90, 2002.
- [16] M. M. Fateh, S. Fateh. A precise robust fuzzy control of robots using voltage control strategy. *International Journal of Automation and Computing*, vol. 10, no. 1, pp. 64–72, 2013.
- [17] Q. Zhu, A. G. Song, T. P. Zhang, Y. Q. Yang. Fuzzy adaptive control of delayed high order nonlinear systems. *International Journal of Automation and Computing*, vol. 9, no. 2, pp. 191–199, 2012.
- [18] J. M. Mendel. *Uncertain Rule-based Fuzzy Logic Systems: Introduction and New Directions*, Upper Saddle River, NJ: Prentice-Hall, 2001.
- [19] L. A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning–1. *Information Sciences*, vol. 8, no. 3, pp. 199–249, 1975.
- [20] D. R. Wu, W. W. Tan. Genetic learning and performance evaluation of interval type-2 fuzzy logic controllers. *Engineering Applications of Artificial Intelligence*, vol. 19, no. 8, pp. 829–841, 2006.
- [21] L. X. Wang. *A Course in Fuzzy Systems and Control*, New York: Prentice Hall, 1996.
- [22] H. A. Hagnas. A hierarchical type-2 fuzzy logic control architecture for autonomous mobile Robots. *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 4, pp. 524–539, 2004.
- [23] T. Ozen, J. M. Garibaldi, S. Musikasawan. Modelling the variation in human decision making. In *Proceedings of IEEE Annual Meeting of the Fuzzy Information*, IEEE, Banff, Alberta, Canada, vol. 2, pp. 617–622, 2000.
- [24] P. Melin, O. Castillo. A new approach for quality control of sound speakers combining type-2 fuzzy logic and fractal theory. In *Proceedings of the IEEE International Conference on Fuzzy Systems*, IEEE, Honolulu, HI, USA, vol. 2, pp. 825–830, 2002.
- [25] V. Mukherjee, S. P. Ghoshal. Intelligent particle swarm optimized fuzzy PID controller for AVR system. *Electric Power Systems Research*, vol. 77, no. 12, pp. 1689–1698, 2007.
- [26] R. Poli, J. Kennedy, T. Blackwell. Particle swarm optimization. *Swarm Intelligent*, vol. 1, no. 1, pp. 33–57, 2007.
- [27] M. M. Fateh, M. M. Zirkohi. Adaptive impedance control of a hydraulic suspension system using particle swarm optimization. *Vehicle System Dynamics*, vol. 49, no. 12, pp. 1951–1965, 2011.
- [28] M. W. Spong, S. Hutchinson, M. Vidyasagar. *Robot Modeling and Control*, New York: Wiley, 2006.
- [29] Q. L. Liang, J. M. Mendel. Interval type-2 fuzzy logic systems: Theory and design. *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 5, pp. 535–550, 2000.
- [30] M. Biglarbegian, W. W. Melek, J. M. Mendel. On the stability of interval type-2 TSK fuzzy logic control systems. *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, vol. 40, no. 3, pp. 798–818, 2010.
- [31] M. M. Fateh. Robust control of flexible-joint robots using voltage control strategy. *Nonlinear Dynamics*, vol. 67, no. 2, pp. 1525–1537, 2012.
- [32] M. M. Fateh. Robust fuzzy control of electrical manipulators. *Journal Intelligent & Robotic Systems*, vol. 60, no. 3–4, pp. 415–434, 2010.



**Majid Moradi Zirkohi** received the B.Sc. degree in electrical engineering from Kerman University, Iran in 2006, and the M.Sc. degree in electrical engineering from Shahrood University of Technology, Iran in 2008. Currently, he is a Ph. D. candidate at Shahrood University of Technology, Iran.

His research interests include soft computing, control theory, and fuzzy logic.

E-mail: majid.moradi.z@gmail.com (Corresponding author)



**Mohammad Mehdi Fateh** received the B.Sc. degree from Isfahan University of Technology, Iran in 1988, the M.Sc. degree in electrical engineering from Tarbiat Modares University, Iran in 1991, and the Ph.D. degree from Southampton University, UK in 2001. He is a full professor at Department of Electrical and Robotic Engineering, Shahrood University of Technology, Iran.

His research interests include nonlinear control, fuzzy control, robotics, intelligent systems, mechatronics and automation.

E-mail: mmfateh@sharoodut.ac.ir



**Mahdi Aliyari Shoorehdeli** received the B.Sc. degree in 2001, the M.Sc. and Ph. D. degrees in electrical engineering from Khaje Nasir Toosi University of Technology, Iran in 2003 and 2008, respectively. He joined Khaje Nasir Toosi University of Technology in 2008, where he is currently an assistant professor of electrical engineering.

His research interests include system identification, intelligent systems and hybrid control.

E-mail: aliyari@eetd.kntu.ac.ir