International Journal of Engineering Research-Online
A Peer Reviewed International Journal
Articles available online http://www.ijoer.in

Vol.3., Issue.5., 2015
(Sept.-Oct.)

RESEARCH ARTICLE

ISSN: 2321-7758

TRAJECTORY TRACKING OF SCARA ROBOT WITH AN ADAPTIVE NEURO-FUZZY CONTROL SCHEME

Abdallah Farrage*, Abd el Badie Sharkawy, Ahmed S. Ali, M-Emad S. Soliman, Hany A. Mohamed.

Mechanical Engineering Department, Assiut University, Assiut, Egypt

Mechatronics Section, Mechanical Engineering Department, Assiut University, 71516 Assiut, Egypt

*The corresponding author, E-mail: abdallahfraag@eng.au.edu.eg

ABSTRACT

The purpose of this paper is to demonstrate the application and implementation of a novel tracking control scheme suitable to real time industrial applications. The **A**daptive **N**euro-**F**uzzy **C**ontroller (ANFC) contains feed-back controller and feed-forward controller. The former is an Adaptive Fuzzy Logic Controller (AFLC) which is used to maintain the closed-loop system stable in the sense of Lyapunov theory, whilst the latter is computed on-line using an Artificial Neural Network (ANN) which has been trained off-line. It is also design to transport the appropriate necessary torque to the robot drivers. On the other hand, the rule base consists of only four rules per each degree of freedom (DOF). As the closed-loop control performance and stability are promoted if more rules are added to the rule base of the FLC. Nevertheless, a compact rule base is used. The role of adaptation of the FLC on the feed-back component is to enhance and amend the tracking performance. The theoretical background of this control algorithm has been published in [1]

The proposed algorithm (ANFC) is implemented on the **S**elective **C**ompliance **A**ssembly **R**obot **A**rm (SCARA robot) as the testing platform. Experimental results are presented for the following three controllers: the conventional PD controller, the FLC and the proposed controller. The controllers are implemented and tested experimentally at the same initial circumstances to compare fairly between their performances. Results show that the investigated proposed controller has outperformed the other controllers.

Keywords: PD-Control, Adaptive Fuzzy Logic Controller (AFLC), Artificial Neural Network (ANN), Adaptive Neuro-Fuzzy Controller (ANFC) , Degree of freedom (DOF).

1. INTRODUCTION

Robot is one of the most important machines for industrial automation. It is used to achieve a high-precision performance; nevertheless, the dynamics governing robot motion still present a challenging control problem. This can be attributed to nonlinearities, uncertainties and strong coupling of the robot dynamics. That is why, feedback controllers have to be applied to accommodate system uncertainties, coupling, and parameters variations.

In fact, there are several types of controllers applied in the field of robotics. The vast majority of these types are the classical controllers, for instance PD and PID controllers, which are widely used in most industrial applications, [2], [3], however their performance are suspicious, once they rely totally on fixed gains which suit neither altered parameters and nor the dynamic motion.

AFLC illustrates particularly robustness both in simulation and experimental applications in dealing with altered unknown parameters [4]–[6]. Adaptive control system has ability to amend its performance to accommodate the optimal behavior. Adaptation of the control systems can be implemented by many different techniques such as gradient sharply descent, ANN, **G**enetic **A**lgorithm and **P**article **S**warm **O**ptimization (PSO).

Eventually, the ANFC is experimental investigated in this tutorial paper. It is implemented on a two DOF; however it can be generalized and implemented on multi degree of freedom. The controller involves adaptive fuzzy feedback controller which accommodate closed loop stability in the sense of Lyapunov and ANN feed-forward controller which identifies the robot dynamics during motion and delivers the ideal and essential torque to robot actuator [7]. Experimental comparison is made with conventional PD controller, FLC in the same conditions to achieve fairly comparison.

The rest of this paper is organized as follows: Section 2 illustrates the system description. Section 3 explains short introduction about ANN and identification process. The control algorithm is presented in section 4. Results and discussion are established in section 5. Finally, section 6 contains our concluding remarks.

2. System Description

The industrial robot (SCARA robot) has been used to test the proposed algorithm experimentally. Generally, there are many efforts which have been made to accommodate the precise control of SCARA robot [8]–[11]. Fig. 1 illustrates a SCARA robot which exists in the Mechatronics Lab, Faculty of Engineering, Assiut University, Egypt. There, indeed, are different applications of a SCARA robot such as pick and place, painting, brushing, and pegging in hole, etc... It has four degrees of freedom (DOF), i.e. waist swivel, elbow swivel, roll and vertical arm. All axes of SCARA are driven by DC electric servo motors.

The DC motors are undergone to a personal computer through the servo power motor drive. The robot is also equipped with feedback position sensor (optical encoder) for each axis. The encoder sends signals back to a high speed computer. The computer compares measured position with the programmed command signals to direct the movement of the robot. The used encoder is an incremental with three tracks, two of them have one thousand holes for each track, whilst, the third track with one hole for the home position. Each encoder is supplied by 5 volts.

The robot is also equipped with velocity sensors (tachometer), two limit switches and a home limit sensor on each axis. Table 1 shows a working range for the four driven axes. Experiments have been performed on only two links, the waist and elbow, which are nominated as $axis\ 1\ \theta_1\ and\ axis\ 2\ \theta_2$. In all experiments, the servo motors have been converted to direct drive DC motors. The feedback signals from the incremental optical encoder and control signals to the motor drive are sent to/from the computer via Programming Control Interface card (PCI) which was programmed by Lab View language. Resolution of the card is 16 bits. The motor driver is MID-7654/7652 Servo Power Motor Drive which was manufactured by the NI Company. It is a complete power amplifier and system interface for use with four axes of simultaneous motion control. The card is well proper for interfacing industrial applications. It can be used for interfacing hardware such as encoders, limit switches, inputs/outputs, and other motion devices.

Fig.1. SCARA robot.

Table1. A working Range for the four driven axes.

Axis	Symbols	Angle	Movement	Function
1	θ_1	270	Waist Swivel	Arm Rotate
2	θ_2	270	Elbow Swivel	Articulate
3	θ_3	360	Roll	Wrist Twist
4	Z	20 cm	Vertical	Up/Down

3. **An artificial Neural Network model and robot identification.**

System identification is a critical part of system analysis and control, and has a great ability of understanding and estimation of the system without need of the modeling and governing equations of the system. Identification is a process of deriving a mathematical model of a predefined part of the world, using observations. System identification has many procedures which are utilized to define the system behavior.

In recent years, artificial neural networks, in fact, have gained a wide attention in control applications because they have the ability to model non-linear systems that can be the most readily exploited in the synthesis of non-linear controllers. A survey of using ANNs in control systems can be found in [12], [13].

This tutorial paper presents the feed-forward torque component. It is determined on-line using artificial neural networks which were trained off-line. In addition, it presents an artificial neural network model for SCARA robot to accommodate path-planning of trajectory tracking. It allows SCARA to navigate and follow its desired goal location. The proposed ANN, in Fig.2, involves three layers "4-10-2" network composed of an input, a hidden, and an output layer. The input layer includes 4 input source neurons$\theta_1, \dot{\theta}_1, \theta_2 \; and \; \dot{\theta}_2$ which are the position and velocity of the waist and elbow links, respectively. The hidden layer consists of 10 neurons. They receive input data from the input layer after multiply them by the values of the synaptic weights which are$w_{i1}, w_{i2}, w_{i3} \; and \; w_{i4}$,[where$i = (1,2, \ldots \ldots \ldots, 10)$ is number of neuron of the hidden layer]. The weights of the output layer, on the other hand, are$v_{1i} \; and \; v_{2i}$. $\tau_{FF_1} \; and \; \tau_{FF_2}$are the feed-forward torque for joint 1 and joint 2, respectively. The employed activation function is hyperbolic tangent for the hidden neurons, whereas it is linear for the output neurons. The neural networks have been trained off-line using experimental data which was previously obtained from open loop control tests for the two links under investigation. The following desired trajectories have been examined for each joint:

$$\theta_{d_1} = \theta_{d_2} = 50\sin(0.056 * 2\pi t) \; , where \; 0 < t < 30 \quad (1)$$

Generally, the proposed neural network can be defined as follows:

$$ANN = \{I, T, w, v, A\} \quad (2)$$

Where I represents the set of input neurons, T denotes the topology of the network including the number of layers and the number of their neurons, $w \; and \; v$ denote the set of synaptic weights values, and A denotes the activation function.

$$I = \{\theta, \dot{\theta}\}$$

$$T = \{L_{in-2}, L_{h0-10}, L_{out-1}\}$$

$$w_{L\,in} = \begin{Bmatrix} w_{1\,1} & w_{2\,1} & \ldots\ldots\ldots & w_{10\,1} \\ w_{1\,2} & w_{2\,2} & \ldots\ldots\ldots & w_{10\,2} \end{Bmatrix}$$

$$v = \{v_{1\,1} v_{1\,2} \; \ldots\ldots\ldots, v_{1\,10}\}$$

$$A = \left\{\frac{[\exp(x) - \exp(-x)]}{[\exp(x) + \exp(-x)]}, 1\right\}$$

The identification error of the system is shown in Fig.3. In this Figure, the identification error of both joint 1 and joint 2 are shown in (a) and(b), respectively. It is the difference between the desired torque and the output torque of neural networks. The aim of the feed-forward component is to deliver on-line the apocopate torque component to the controller which is suitable to carry the desired trajectory equation (1). A generalized model for the robot which can be implemented for any trajectory is out of scope of this paper. The best training performance of ANN is shown in Fig.4.

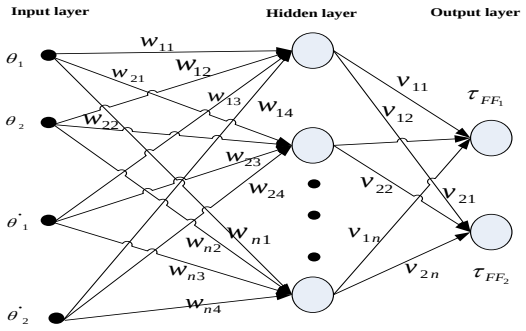


Fig.2. Feedforward Neural Network of the system.

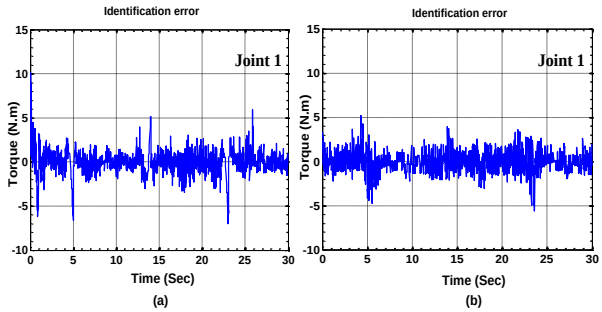


Fig.3. Identification error.

4. The control algorithm

The robotic system is one of the greatest tools to test robustness of control algorithm. The nonlinearities, changed friction coefficients, and strong coupling of the robot dynamics present a big challenging control problem.

Three algorithms, in this tutorial paper, had been applied, experimentally. These algorithms are PD, FLC and the proposed controller. The parameters of joints are considered $\theta = [\theta_1\ \theta_2]$ where $\theta_1$is the angle of joint 1, and θ_2 stands for joint 2. We consider the state variables as $\theta(t)$ and $\dot{\theta}(t)$which are usually available as feedback signals.

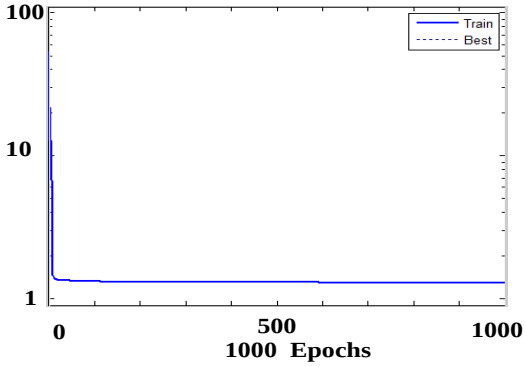


Fig.4. The best training performance.

The robotic system is one of the greatest tools to test robustness of control algorithm. The nonlinearities, changed friction coefficients, and strong coupling of the robot dynamics present a big challenging control problem.

Three algorithms, in this tutorial paper, had been applied, experimentally. These algorithms are PD, FLC and the proposed controller. The parameters of joints are considered $\theta = [\theta_1\ \theta_2]$ where $\theta_1$is the angle of joint 1, and θ_2 stands for joint 2. We consider the state variables as $\theta(t)$ and $\dot{\theta}(t)$which are usually available as feedback signals.

Define the tracking error vectors $e(t)$ and $\dot{e}(t)$ as:

$$e(t) = \theta(t) - \theta_d(t) \quad (3)$$

$$\dot{e}(t) = \dot{\theta}(t) - \dot{\theta}_d(t) \quad (4)$$

whereθ_d and $\dot{\theta}_d$ are vectors of the desired joint position and velocity, respectively.

4.1 PD Control

The performance of the classical control depends totally on its gains, as the output torque of proportional-derivative (PD) controller is defined by:

$$U(t) = K_P e(t) + K_d \dot{e}(t) \quad (5)$$

where, K_P and K_d are $2x2$ positive definite diagonal matrices called the proportional and the derivative gain matrices of the controller, respectively. In spite of its simplicity, a feasible problem associated with PD-control is the constant control gains; in addition, once the values of these gains exceed their critical values, the system becomes unstable. That is why the performance of PD-controller is restricted with these values of gains which are usually experimentally determined. The block diagram of the PD controller is shown in Fig.5.

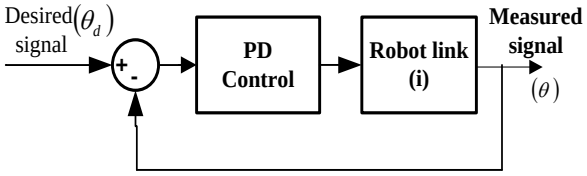


Fig.5. Block diagram of PD Controller.

4.2 Fuzzy logic control (FLC)

The performance of fuzzy logic control is dependent on its inference rules dramatically. The number of used rules, in this paper, is four rules only. They can be derived as follows [1].

Consider e_i and $\dot{e}_i$ are the error and the change in error of the system, and i is the DOF. The Lyapunov direct method can be applied to accommodate the stability of the system. Consider the following Lypunov candidate function:

$$V = \frac{1}{2}(e^T e + \dot{e}^T \dot{e}) \quad (6)$$

By differentiating with respect to time gives

$$\dot{V}_i = e_i \dot{e}_i + \dot{e}_i \ddot{e}_i \quad (7)$$

To achieve asymptotic stability, it is required to find u_{FB_i} so that

$$\dot{V}_i = e_i \dot{e}_i + \dot{e}_i \ddot{e}_i \leq 0 \quad (8)$$

In some neighborhood of the equilibrium of (6). Taking the control signal $u_{FB}$to be proportional to$\ddot{e}$ so the previous equation can be rewritten as:

$\dot{V}_i = e_i\dot{e}_i + \alpha_i\dot{e}_i u_{FB_i}$ (9)

where, α_i is positive constant and u is the control signal.

Sufficient conditions for the previous equation can be stated as follows.

- If for $i \in [1, \ldots, n]e_i$ and $\dot{e}_i$ have opposite signs then $\alpha_i u_{FB_i}$ must to be zero;
- If for $i \in [1, \ldots, n]e_i$ and $\dot{e}_i$ are both positive then $\alpha_i u_{FB_i}$ must to be negative;
- If for $i \in [1, \ldots, n]e_i$ and $\dot{e}_i$ are both negative then $\alpha_i u_{FB_i}$ must to be positive;

One can easily obtain the four rules listed below in table 2. In this table P, N, denote positive and negative error, respectively. u_P ,u_N, and $u_Z$are positive, negative, and zero control signal respectively. These rules are simple and satisfy the conditions of stability.

Table 2. Fuzzy rules for the fuzzy feedback controller.

$\dot{e}_i$ / e_i	P	N
P	u_N	u_Z
N	u_Z	u_P

To complete the design, we must specify the fuzzy system with which the fuzzy feedback computes the control signal. The Gaussian membership defining the linguistic terms in the rue base is chosen as follows:

$$\mu_{positive}(x) = G(x, a) = e^{-(x-a)^2}$$

$$\mu_{negative}(x) = G(x, -a)$$

$$\mu_{Zero}(x) = G(x, 0)$$

where, $a > 0$.

The above four rules can be represented by the following mathematical expression

$$u_{FB_i} = \frac{G(e_i, a_{1i})(-k_i) + G(e_i, -a_{1i})(k_i)}{G(e_i, a_{1i}) + G(e_i, -a_{1i})} + \frac{G(\dot{e}_i, a_{2i})(-k_i) + G(\dot{e}_i, -a_{2i})(k_i)}{G(\dot{e}_i, a_{2i}) + G(\dot{e}_i, -a_{2i})}$$

In more details

$$u_{FB_i} = -k_i\left[\frac{exp(-(e_i - a_{1i})^2) - exp(-(e_i + a_{1i})^2)}{exp(-(e_i - a_{1i})^2) + exp(-(e_i + a_{1i})^2)} + \frac{exp(-(e_i - a_{2i})^2) - exp(-(e_i + a_{2i})^2)}{exp(-(e_i - a_{2i})^2) + exp(-(e_i + a_{2i})^2)}\right]$$

From which

$$u_{FB_i} = -k_i\left[\frac{exp(2a_{1i}e_i) - exp(-2a_{1i}e_i)}{exp(2a_{1i}e_i) + exp(-2a_{1i}e_i)} + \frac{exp(2a_{2i}\dot{e}_i) - exp(-2a_{2i}\dot{e}_i)}{exp(2a_{2i}\dot{e}_i) + exp(-2a_{2i}\dot{e}_i)}\right]$$

wherea_{1i} and a_{2i} are positive constants. The control law becomes

$u_{FB_i} = -k_i[\tanh(2a_{1i}e_i) + \tanh(2a_{2i}\dot{e}_i)]$,

$i = 1, \ldots, n$ (10)

This equation gives the feedback control signal needed to stabilize the system. The inputs of this equation are only $e_i$and $\dot{e}_i$, and $u_{FB_i}$is the control input for each joint$i$. The block diagram of the FLC is shown in Fig.6. The features of this control law are

- This control law is a special case of the fuzzy systems.
- Only three parameters per each DOF need to be tuned, namely, they are k_i, a_{1i}, *and* a_{2i}This controller is inherently bounded since $|tanh(x)| \leq 1$.
- Each joint has independent control input u_{FB_i}, $i = 1,2, \ldots, n$.

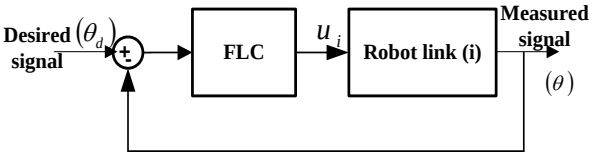
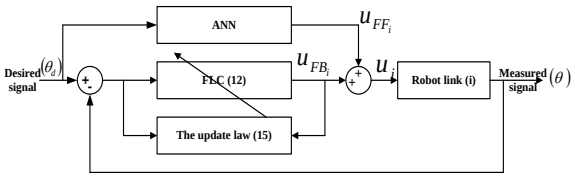


Fig.6. block diagram of the FLC Controller.

4.3 **The proposed controller**

The proposed control involves feedback and feed-forward torque components. The feed-back torque component is produced from adaptive FLC which maintains the stability of the system. The feed-forward torque is computed on-line using an artificial neural network (ANN) as mentioned above in Section 3. It is designed to deliver the ideal torque to the robot derivers. The addition of this feed-forward component, in fact, improves the performance[14]. The inputs of feed-forward component are□, and □ for each joint and the output is the torque □, □ = *1,2*. The overall closed-loop control system is shown in Fig.7. In this Figure, □ is the value of the

feed-forward torque u_{FB_i} is the value of torque which is produced by the adaptive FLC, and $u_i$is the value of the total input torque to the robot driver.



5. Experimental Results.

In this section, the experiments conducted using three different controllers are presented. These controllers are conventional PD controller, Fuzzy Logic Controller (FLC) and the proposed controller. The whole experimental system includes the host computer, Data Acquisition, and the SCARA robot. In all the experiments, the three controllers are implemented with the same initial position error equal 10^o, i.e. $q(0) = [10^o - 10^o]^T$to investigate of the strength and weakness of each one. This condition undergoes an initial position error, $e = [0.175 \ -0.175]^T$ radians. The desired sinusoidal trajectory, amplitude 50^o or $\left(5 * \frac{\pi}{18}\right) radian$ and frequency $0.056\, cycle/S$, was applied for three types of controllers$[\theta_d = 50 * sin(0.056 * 2\pi t)]$.

From one hand, the trajectory tracking of conventional PD algorithm and FLC is illustrated in Fig.8 and Fig.9, respectively. The desired and measured trajectories of both joint 1 and joint 2 are shown in $(a) and\ (b)$. The solid and dash lines, in both Figures, describe the desired and actual angles, respectively. It is interesting to note that the system is more stable and the tracking error is well reasonable in FLC rather than conventional algorithm. This is true in Fig.10, which shows the error signals of both algorithms, in addition, the measured torque of two algorithms can be found in Fig.11 and Fig.12.

On the other hand, the trajectory tracking of the proposed controller in Fig.13 shows that the actual trajectory is more identical with the desired one than PD and FLC. This can be evident in Fig.14 as the error signals are more closed to zero for each joint.

The torque signal of the proposed algorithm in Fig.15 was built from both torque of adaptive fuzzy in Fig.16 and feed-forward torque. The latter torque in Fig.17 is resulted from robot identification. In contrast, up-dated gain of proposed algorithm which is mentioned in (15) is demonstrated in Fig.18. It is more interesting to see that Fig.19 and Fig.20 display the error signals of the three controllers collected for joint 1 and joint 2, respectively to make obvious and fair comparison between them.

In this paper, two strategies of the tracking error are presented to compare fairly the performance of the three controllers. The first method is the scalar value Root Mean Square error (RMS) which is defined as

$$RMS = \sqrt{\frac{1}{T_f}\sum_{j=1}^{N}[\theta_d(j) - \theta(j)]^2 \Delta T}$$

Where $\theta_d and\ \theta$ are the desired and measured trajectories, respectively.T_f is the final time which is equal $N * \Delta T$. The performance of these criteria for three controllers is presented in Fig.19. This figure shows that values of RMS errors gradually decrease from PD to proposed controller.in both joints. In contrast the second strategy is known as the maximum absolute value of the tracking error which is applied after two second from the starting time. It is defined as$e_{max} = \max_{0 \le j \le N}|\theta_d(j) - \theta_m(j)|$. The performance of these criteria is shown in Fig.20. It is more interesting to note that there is a dramatically decline between the values of PD algorithm and other algorithms.

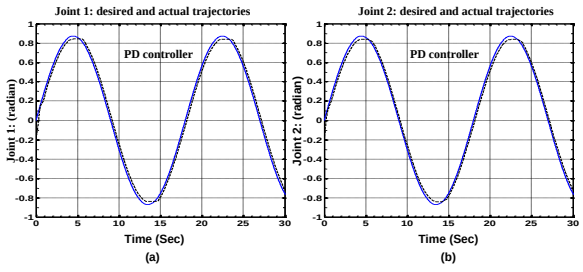


Fig.8. The performance of the conventional PD controller.

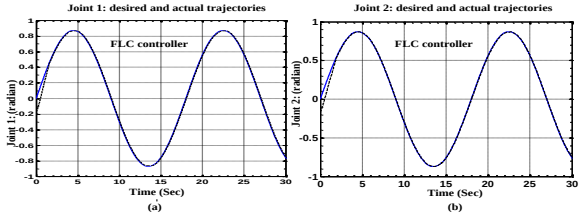


Fig.9. The performance of the FLC controller.

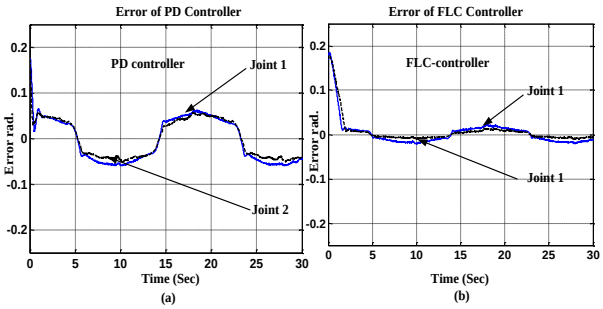


Fig.10. The error performance.

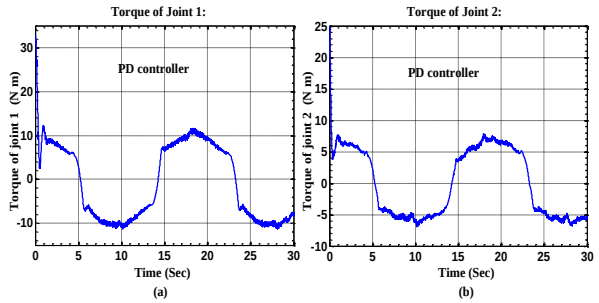


Fig.11. The torque signal of PD controller.

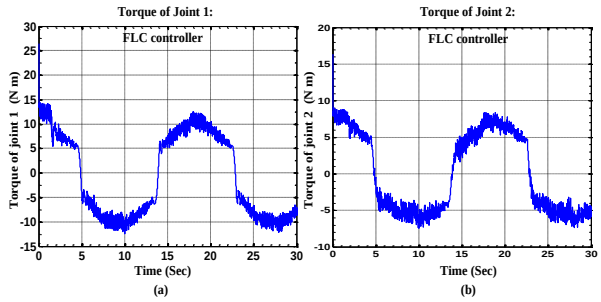


Fig.12. The torque signal of the FLC controller.

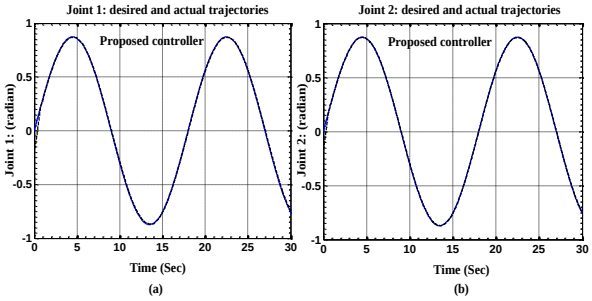


Fig.13. The performance of the proposed controller.

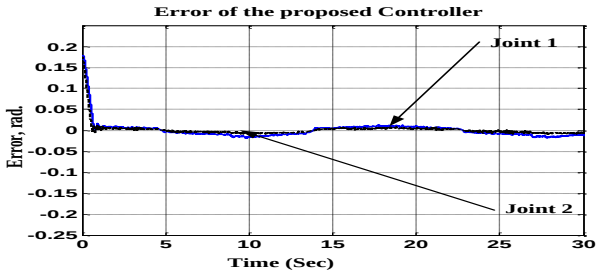


Fig.14. The error performance of the proposed controller.

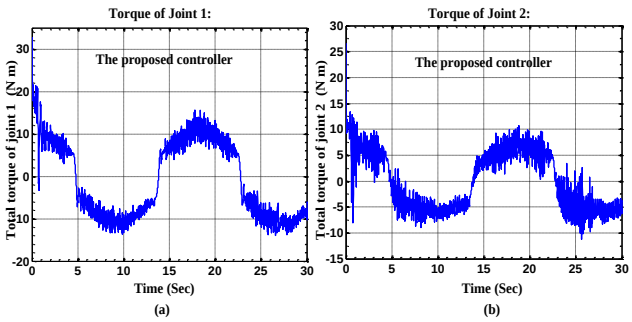


Fig.15. The total torque signal of the proposed algorithm.

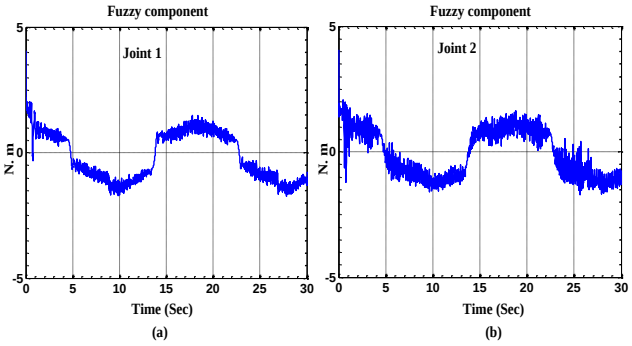


Fig.16. The value of fuzzy component of the proposed algorithm.

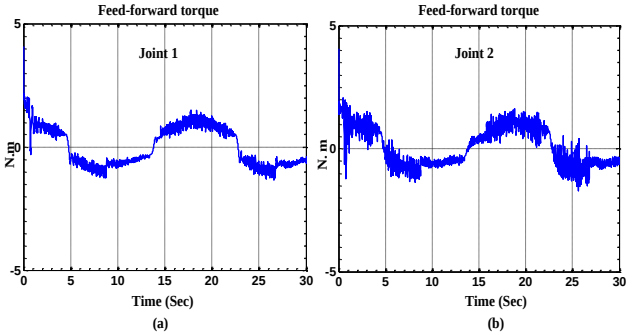


Fig.17. The value of feed-forward component of the proposed algorithm.

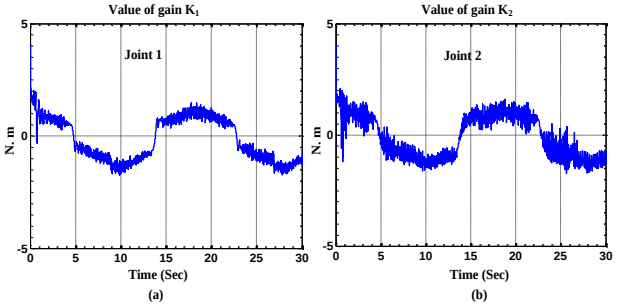


Fig.18. The value of up-date gain k_i of the adaptive controller.

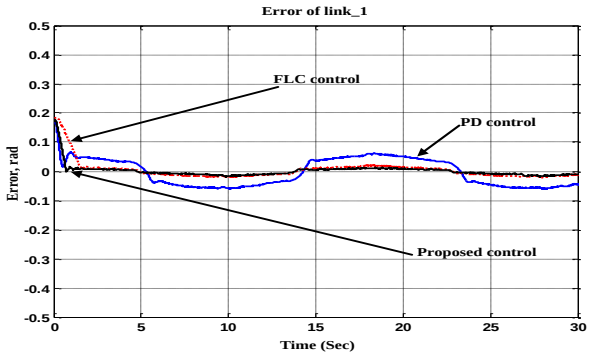


Fig.19. The error performance of joint 1.

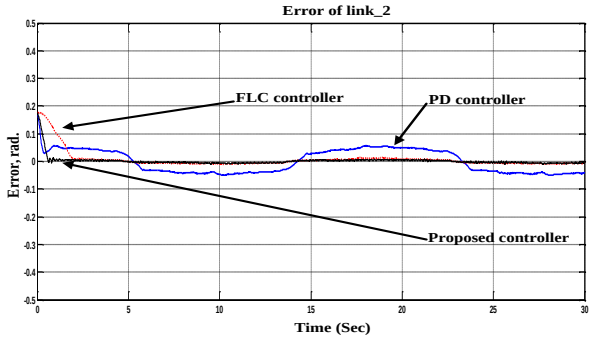


Fig.20. The error performance of joint 2.

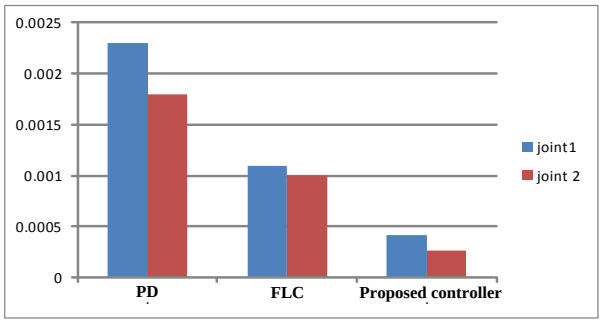


Fig.21. The RMS error for the three algorithms in radian.

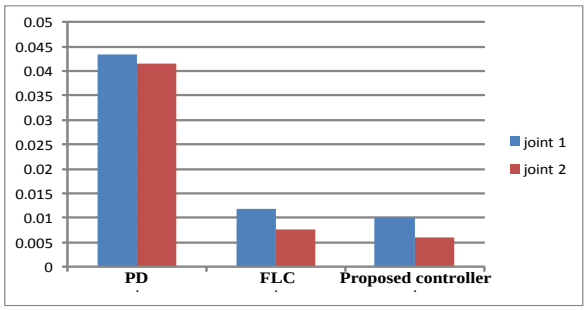


Fig.22. The maximum error for the three algorithms in radian.

6. Conclusion

The proposed controller suitable for industrial applications has been successfully designed, developed, and implemented in this paper. We have implemented the Lyapunov theory to get rules of the fuzzy controller so that the system is stable in the sense of Lyapunov. Furthermore, the performance index has applied to tune the gain of the fuzzy law. Also we used ANN to identify the SCARA Robot. The proposed algorithm is very simple and its implementation does not need more complicated computationally tasks. The experiments are implemented on only two joints, but the proposed algorithm can be extended to (n) number of link robots. The proposed algorithm can be used in different control industrial applications.

Experimental results show that the proposed algorithm has been successful in representing the nonlinearity system. They also show that the system becomes more stable and the tracking error becomes near to zero. The initial error is used to test the robustness of the system. Relative to Conventional PD-controller and FLC, the proposed algorithm achieves the best performance and achieves the greatest results.

REFERENCES

[1]. A. B. Sharkawy, "*A computationally efficient fuzzy control scheme for a class of MIMO systems*" *ALEXANDRIA Eng. J.*, pp. 1–12, 2013.

[2]. I. Cervantes and J. Alvarez-ramirez, "*On the PID tracking control of robot manipulators," Syst. Control Lett.*, vol. 42, pp. 37–46, 2001.

[3]. J. Alvarez-ramirez, I. Cervantes, and R. Kelly, "*PID regulation of robot manipulators : stability and performance," Syst. Control Lett.*, vol. 41, pp. 73–83, 2000.

[4]. S. Jagannathan, M. W. Vandegrift, and F. L. Lewis, "*Adaptive fuzzy logic control of discrete-time dynamical systems," Automatica*, vol. 36, pp. 229–241, 2000.

[5]. K. Chiou and S. Huang, "*An adaptive fuzzy controller for robot manipulators," Mechatronics*, vol. 15, pp. 151–177, 2005.

[6]. J. P. Perez, R. Soto, A. Flores, F. Rodriguez, and L. Meza, "*Trajectory Tracking Error Using PID Control Law for Two-Link Robot Manipulator via Adaptive Neural Networks," Procedia Technol.*, vol. 3, no. 81, pp. 139–146, 2012.

[7]. F. Cus, U. Zuperl, and J. Balic, "*Combined feedforward and feedback control of end*

milling system," *J. Achiev. Mater. manfacturing Eng.*, vol. 45, no. 1, pp. 79–88, 2011.

[8]. F. Passold, P. Fundo, and M. R. Stemmer, "*Force Control of a Scara Robot using Neural Networks*" Federal University of Santa Catarina, *Fourth Int. Work. Robot motion Control*, pp. 247–252, 2004.

[9]. N. A. Shiltagh and D. A. A. Kadeer, "*MODIFIED TRAINING METHOD FOR FEEDFORWARD NEURAL NETWORKS AND ITS APPLICATION in 4-LINK SCARA ROBOT IDENTIFICATION*," *J. Eng.*, vol. 17, no. 5, pp. 1335–1344, 2011.

[10]. S. Yamacli and H. Canbolat, "*Simulation Modelling Practice and Theory Simulation of a SCARA robot with PD and learning controllers*," *Simul.Model.Pract.Theory*, vol. 16, no. 9, pp. 1477–1487, 2008.

[11]. A. Marwan, F. Nagi, K. S. M. Sahari, S. Hanim, and I. Fadi, "*On-Line Adaptive Fuzzy Switching Controller for SCARA Robot*," *Syst. Control Lett.*, vol. 6, no. 11, pp. 404–416, 2011.

[12]. S. Paper, "*Neural Networks for Control Systems A Survey*," *Int. Fed. Autom. Control*, vol. 28, no. 6, pp. 1083–1112, 1992.

[13]. H. D. Patiño, R. Carelli, S. Member, and B. R. Kuchen, "*Neural Networks for Advanced Control of Robot Manipulators*," *IEEE Trans. neural networks*, vol. 13, no. 2, pp. 343–354, 2002.

[14]. S. Kolyubin, D. Efimov, V. Nikiforov, H. B. A, and P. Plaza, "*Control of Nonlinear Systems Using Multiple Model Black-Box Identification*," vol. 2013, 2013.