ADAPTIVE FUZZY CONTROL OF
A STEWART PLATFORM MANIPULATOR

Ayman A. ALY*, Hidetoshi OHUCHI**

*Graduate School, Yamanashi University
(ayman_aly@ms.yamanashi.ac.jp)
**Yamanashi University, Faculty of Engineering, 4-3-11,
Takeda, Kofu, Yamanashi, 400-8511, Japan
(ohuchi@ms.yamanashi.ac.jp)

ABSTRACT

The Stewart platform is one example of a parallel manipulator with high force to weight ratio and fine positioning accuracy far exceeding those of a conventional serial-link arm. It is basically a closed-link type robot manipulator having 6 degrees of freedom. In this paper the implementation of an adaptive control scheme based on fuzzy logic theory is used to control the motion of a Stewart platform manipulator. The inverse kinematics is analysed and six individual controllers are implemented in the actuators coordinates. An experimental study is conducted to evaluate the performance of the proposed control scheme implemented to control the manipulator to track a defined path. Experimental results show that the proposed control policy provides superior tracking capability as compared to the fixed-gain controllers.

KEY WORDS

Electro-hydraulic servo drives, Stewart platform, Fuzzy adaptive control, Model reference, Learning.

INTRODUCTION

Parallel manipulators can be found in many applications, such as flight simulators, adjustable articulated trusses, mining machines, pointing devices and walking machines. Recently, the Stewart platform manipulator has also been developed as a high-speed, high-precision and multi-DOF machining centre.

These parallel manipulators possess the advantages of high stiffness, low inertia, low accumulation of joint errors and large payload capacity. However, they suffer the problems of relatively small useful workspace and design difficulties. Furthermore, their direct kinematics is a very difficult problem.

As shown in Figure 1, a parallel manipulator typically consists of a moving platform that is connected to a fixed base by several limbs or legs. The number of the legs is equal to the number of the freedom such that every limb is controlled by one actuator. Because all of the actuators can be mounted on the fixed base, parallel manipulators tend to have a large load-carrying capacity.

The researches in the parallel manipulators are mainly concern in three studying areas: closed form solution for the forward and inverse kinematics, derivation of dynamics equations and designing an adequate controller [1] [2] [3].

In this paper, the inverse kinematics of the Stewart platform is analysed on the base of geometric configuration and a 6-DOF trajectory tracking control system is implemented. Considering the robustness against the nonlinearity of the system parameters and the resultant accuracy with the supply pressure change, FMRLC (Fuzzy Model Reference Learning Control) scheme is adopted for tracking the given referred trajectory.

INVERSE KINEMATICS

For the purpose of analysis, two Cartesian coordinate systems, frames $A(x,y,z)$ and $B(u,v,w)$ as shown in Figure 1, are attached to the fixed base and the moving platform, respectively. The transformation from the moving platform to the fixed base can be described by the position vector $\boldsymbol{P}$ of the centroid P and the rotation matrix ${}^{A}R_{B}$ of the moving platform. Let $\overline{u}$, $\overline{v}$, and $\overline{w}$ be three unit vectors defined along the u,v, and w axes of the moving coordinate system; then the rotation matrix can be written as

$${}^{A}R_{B} = \left[\overline{u}\ \overline{v}\ \overline{w}\right] \qquad (1)$$

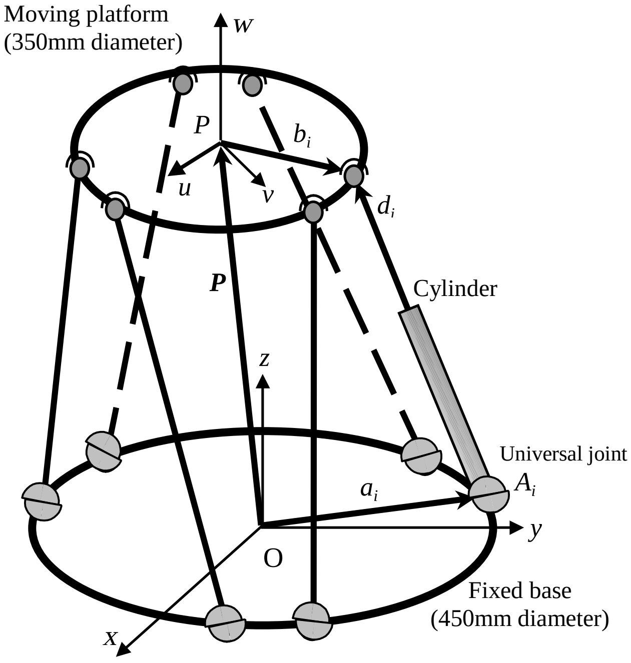


Figure 1 Spatial 6-dof, Stewart platform manipulator.

Note that the elements of ${}^{A}R_{B}$ must satisfy the orthogonal conditions. As shown in Figure1, let:

$$a_i = [a_{ix}, a_{iy}, a_{iz}]^T \text{ and } {}^{B}b_i = [b_{iu}, b_{iv}, b_{iw}]^T$$

be the position vectors of points A_i and B_i in the coordinate frames A and B, respectively. We can write a vector-loop equation for the i^{th} limb of the manipulator as follows:

$$\overline{A_i B_i} = P + {}^{A}R_{B}\,{}^{B}b_i - a_i \qquad (2)$$

The length of the i^{th} limb is obtained by taking the inner product of the vector $\overline{A_i B_i}$ with itself:

$$d_i^2 = [P + {}^{A}R_{B}\,{}^{B}b_i - a_i] \bullet [P + {}^{A}R_{B}\,{}^{B}b_i - a_i],$$

$$\textit{for } i=1,2,\ldots,6 \qquad (3)$$

where d_i denotes the length of the i^{th} limb. For the inverse kinematics problem, the position vector $\boldsymbol{P}$ and rotation matrix ${}^{A}R_{B}$ of frame B with respect to A are given and the limb lengths d_i are to be find. Expanding Eq. (3) and by taking the square root we obtain:

$$d_i = \sqrt{P^T P + {}^{B}b_i^T\,{}^{B}b_i + a_i^T a_i + 2P^T\,{}^{A}R_B\,{}^{B}b_i - 2P^T a_i - 2[{}^{A}R_B\,{}^{B}b_i]^T a_i} \qquad (4)$$

This yields six equations describing the location of the moving platform with respect to the fixed base. When the solution of d_i becomes a complex number, the location of the moving platform is not reachable.

Inverse kinematics singularities cannot occur within the workspace of the manipulator. However, they can occur at the workspace boundary, where one or more limbs are in fully stretched or retracted positions.

Joint space schemes as shown in Figure 2 are usually the easiest to compute and the calculation time is shorten by implementing the inverse kinematics calculation outside the control loop.

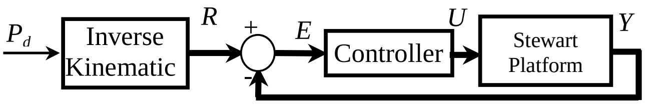


Figure 2 Joint-based control scheme.

Where P_d is the desired Cartesian space coordinate, $R=[r_1,r_2,\ldots\ldots..,r_6]^T$ are the desired joint space coordinates, $E=[\ e_1,e_2,\ldots\ldots..,e_6]^T$ are the errors between the desired and the actual positions, $U=[u_1,u_2,\ldots\ldots..,u_6]^T$ are the control signals, and $Y=[\ y_1,y_2,\ldots\ldots..,y_6]^T$ are the measured positions.

This control scheme generally results in a controller running at a higher sampling frequency than Cartesian-based controller. That would also, in general, increase the stability and disturbance rejection capabilities of the system.

DESIGN OF MODEL REFERENCE ADAPTIVE CONTROLLER

Because of the fact that the closed-form solution of forward kinematics is hard to obtain for a Stewart platform structure, six individual controllers are implemented in the actuators coordinates. By using the inverse kinematics analysis, a model reference learning control scheme is utilized.

A direct tracking control architecture for class of continuous-time non-linear dynamic systems has been proposed [4]. The simulation results verified the effectiveness of the proposed control algorithm. In [5] a design method of Fuzzy control systems depending on trial and error has been presented, and effective and convenient support tools for the study and design of Fuzzy control systems have been introduced. A self-learning Fuzzy logic system for (MIMO) plants has been introduced [6], where a plant model is not required for training. Instead, training is guided by observations of plant responses to inputs.

J. R. Layne and K.M. Passino introduced a theoretical study for controlling cargo ship steering by using FMRLC that depends on changing the final controller output by adding direct correction for the final value based on defined model [7]. They extended their theoretical study for the applications to two degrees of freedom manipulator [8].

However the previous studies based on FMRLC are mainly theoretical, in this article we attempt to study the practical application of FMRLC to an electro-hydraulic system, where the centers of the output member ship functions are adapted by using learning mechanism to make the system follow the defined reference. A learning control system is designed so that its “learning controller” has the ability to improve the performance of the closed loop system by utilizing feedback information from the plant.

The functional block diagram of the FMRLC for an axis is shown in Figure 3. It has three main parts: the fuzzy controller to be tuned, the reference model and the learning mechanism.

The FMRLC uses the learning mechanism to observe numerical data from a fuzzy control system. Using this numerical data, it characterizes the current performance of the fuzzy control system and automatically adjusts the fuzzy controller so that the system performance can meet some given objectives.

THE FUZZY CONTROLLER AND FUZZY REASONING

The fuzzy logic controller includes three important steps: fuzzification, fuzzy reasoning (decision making) and defuzzification. The inputs to the fuzzy controller are the error $e_i(kT)$ and the rate of error change $c_i(kT)$.

The basic operation of the inference process is to determine the values of the controller output based on the contributions of each rule in the rule base. One method of storing the rule base is the use of the Macvicar-Whelan control matrix (Table 1). Each element of the matrix describes a rule of the form:

$$If\ \ e_i\ is\ E_i^j\ and\ \ c_i\ is\ C_i^l\ Then\ \ u_i\ is\ U_i^m \qquad (5)$$

where E_i^j, C_i^l and U_i^m are the j^{th}, l^{th} and m^{th} linguistic value associated with $e_i(kT)$, $c_i(kT)$ and $u_i(kT)$, respectively.

We must choose initial values for each of the output membership functions. For example, for an output universe of discourse [-1, 1] we could choose triangular shaped membership functions with width of 0.4 and centers at zero.

THE REFERENCE MODEL

The goal of the FMRLC is to make the closed-loop system behave like a given “reference model”. In our experiments, a reference model of the second order scheme is used. Here we would like the system to respond faster without overshoots, so the pole of the closed-loop system should be further into the left-half plane and have critical damping value. In the Laplace domain, the model form is represented as follows:

$$G(s) = {w_m^2} \Big/ \left(s^2 + 2*\zeta*w_m s + w_m^2\right) \qquad (6)$$

where ζ is the damping ratio and w_m is the undamped natural frequency.

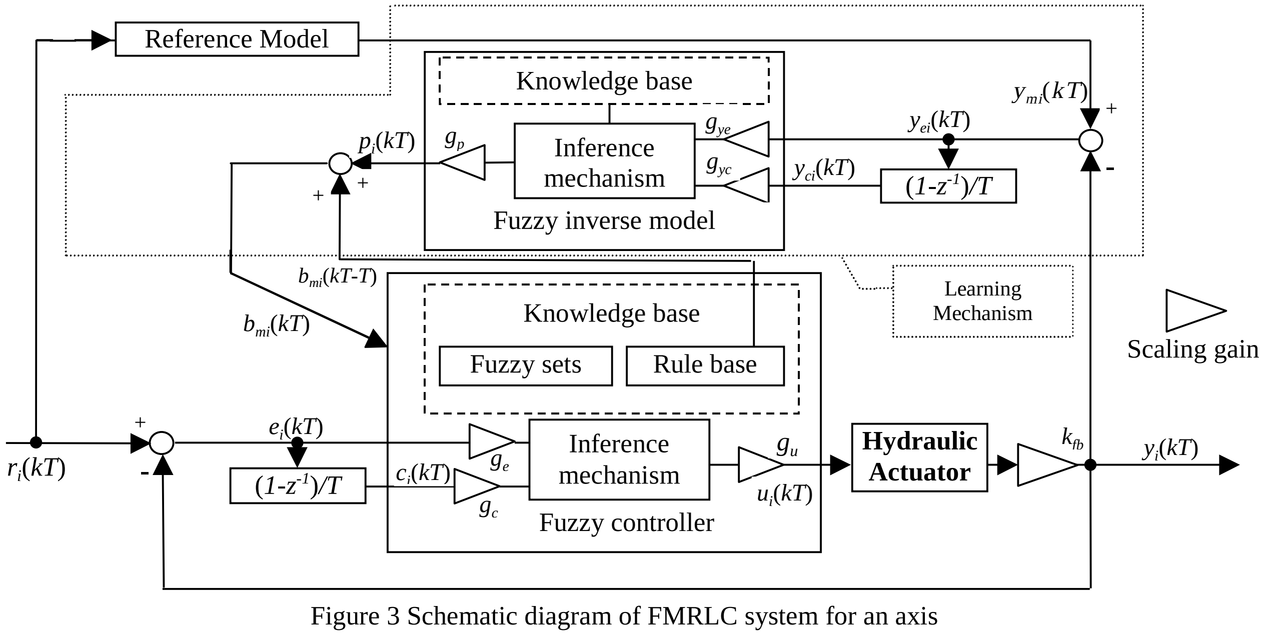


Figure 3 Schematic diagram of FMRLC system for an axis

LEARNING MECHANISM

The learning mechanism tunes the rule base of the direct fuzzy controller by changing the centers of the output membership functions so that the closed loop system behaves like the reference model. These rule base modifications are made by observing data from the controlled process, the reference model and the fuzzy controller. The learning mechanism consists of two parts: a “fuzzy inverse model” and a “knowledge base modifier”.

Table 1 Macvicar-Whelan fuzzy rule matrixes

c_i / e_i	**NB**	**NS**	**ZO**	**PS**	**PB**
NB	NB	NB	NB	NS	ZO
NS	NB	NB	NS	ZO	PS
ZO	NB	NS	ZO	PS	PB
PS	NS	ZO	PS	PB	PB
PB	ZO	PS	PB	PB	PB

NB (**N**egative **B**ig), **NS** (**N**egative **S**mall), **ZO** (**Z**ero), **PS** (**P**ositive **S**mall) and **PB** (**P**ositive **B**ig).

Fuzzy inverse model

Similar to the fuzzy controller, the fuzzy inverse model shown in Figure 3 produces $p_i(kT)$ by using the fuzzy inference mechanism with the rule of Table1.

Given that y_{ei} and y_{ci} are inputs to the fuzzy inverse model, the rule base for the fuzzy inverse model contains rules of the form:

$$If \quad y_{ei} \text{ is } Y_{ei}^{j} \quad and \quad y_{ci} \text{ is } Y_{ci}^{l} \quad Then \quad p_i \text{ is } P_i^{m} \qquad (7)$$

Where Y_{ei}^{j} and Y_{ci}^{l} denote linguistic values and P_i^{m} denotes the linguistic value associated with the m^{th} output fuzzy set. The value of $p_i(kT)$ gives the necessary changes in the plant input to reduce the plant error.

Knowledge base modifier

Given the necessary changes in the input to the plant, to force the error y_{ei} to zero, the knowledge base modifier changes the rule base of the fuzzy controller so that the control action $u_i(kT)$ will be modified by the action of $p_i(kT)$ as a result of shifting the centers of the membership functions.

By modifying the fuzzy controller’s knowledge base, we may force the fuzzy controller to produce a desired output. Let b_{mi} denote the center of the symmetric membership function associated with U_i^{m}, and the knowledge base modification is performed by shifting the values of b_{mi}.

$$b_{mi}(kT) = b_{mi}(kT-T) + p_i(kT) \qquad (8)$$

This modification affects only on the centers of membership functions that are used in the current fuzzy reasoning.

EXPERIMNTAL RESULTS

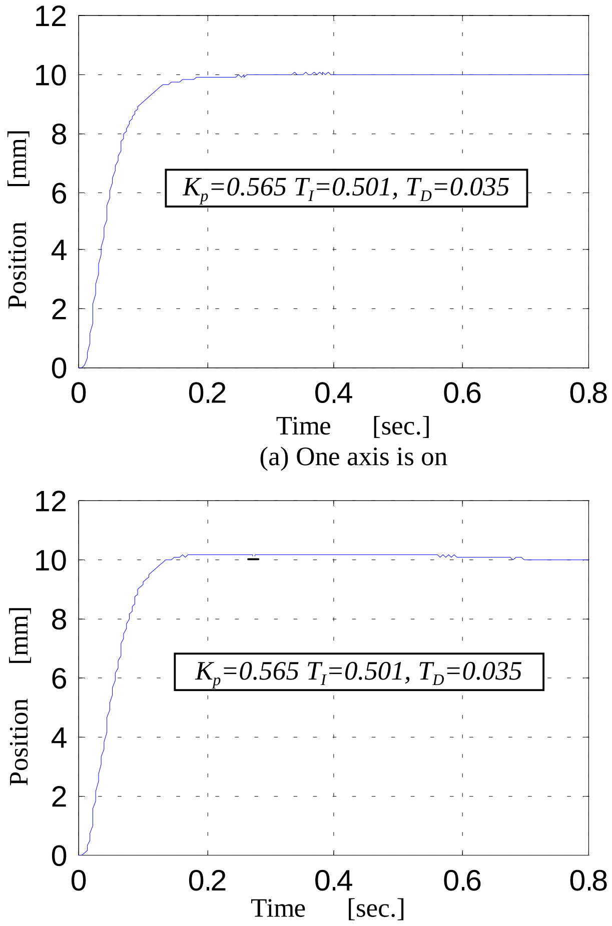
The real time implementation was carried out using the sampling time T of 2msec and the same reference trajectory input with 10kg payload and 2.5MPa supply pressure. The robot limbs are hydraulic cylinders of 20mm inners diameter, 15mm rods diameter and 98mm full strokes. The control algorithm is implemented by a personal computer, based on a 1.66 GHz Pentium processor.

An experimental test was conducted, at first, by using a fixed gain classical controller (PID) as follows,

$$D[z] = K_P\left(1 + \frac{T\ z}{T_I\,(z-1)} + \frac{T_D\,(z-1)}{T\ z}\right) \qquad (9)$$

where the controller parameters [K_P, $T_{I,}$ T_D] were adjusted manually to get faster response without overshoots and with zero steady state error. The controller parameters were chosen for each axis whenever the other axes were kept stay.

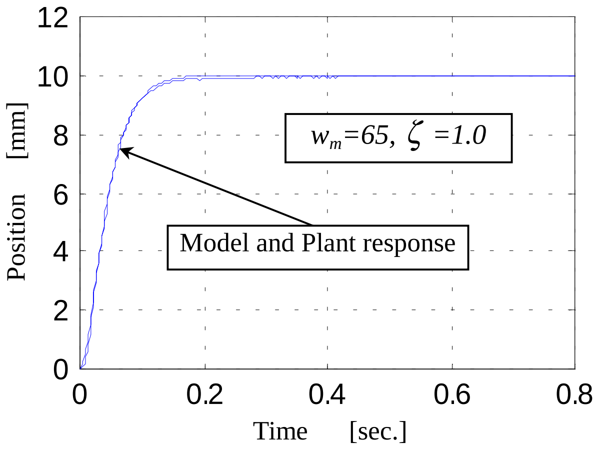
The reference input of the manipulator at the teaching center point (TCP) was a step input with amplitude of 10mm. Figure 4.a shows the response of an axis based on PID controller, while the other axes were off. The system response was coincident with the reference after 0.33sec.



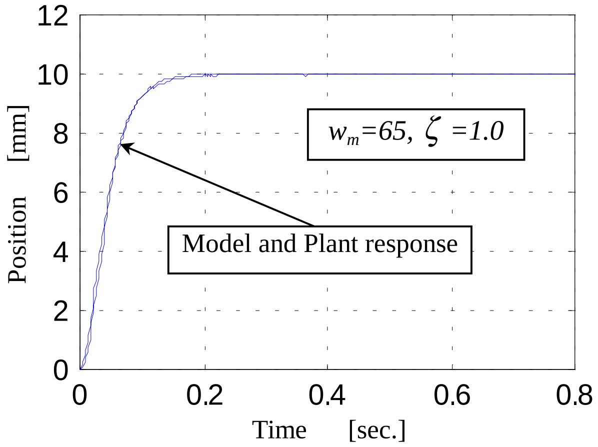
(a) One axis is on

(b) six axes are on

Figure 4 An axis step response based on PID Control.



(a) One axis is on



(b) all axes are on

Figure 5 An axis step response based on FMRLC.

In Figure 4.b the test was repeated for the same axis while the same inputs were fed to all of the other axes. It is noticed that the system response arrived at the steady state after about 0.7sec and small overshoot appeared. As we tried to improve the response, the confliction effect of the gain changes limited the ability of improving. For example, if we want to increase the response speed by increasing the proportional or the derivative gains, the stability gets worse, and if we want to improve it by increasing the integral gain the speed becomes slower and so on. It is clear the difference between the two responses is according to the axes coupling effects and the change of the supply pressure.

The system response based on FMRLC is introduced in Figure 5 for the same axis under the same reference input. As before, the other axes are off in Figure 5 (a) and on in Figure 5 (b). It is noticed the controller has the ability to provide robustness with faster response (the response arrived at the steady state region in 0.18sec.).

The pressure changes are shown in Figures 6 and 7 with the same conditions of Figures 4.b and 5.b. When only one axis was driven based on PID controller, the supply pressure change was within 0.5MPa and when all axes were on it was more than 1.0MPa. While, with FMRLC (when only one axis was on) the pressure change was within 0.38MPa and when all axes were on it was about 0.76MPa.

Although, the inertia difference of the moving parts and the large pressure change happened, FMRLC can regulate the system to keep the response behave like the model. This is due to the fact that the learning ability of the controller can eliminate the error between the plant response and the model response. In Figure 8, the modification by the adaptation signals $p_i(kT)$ is shown for the case of Figure 5 (b), where the adaptation continued until each of the axes responses became coincident with the reference inputs.

For the testing of motion control, an example of circle reference input to the TCP with 40 mm diameter was applied. The TCP response and the axes responses based on the proposed controller is presented in Figures 9 and 10 respectively. It is clear the ability of the controller to force the system motion to follow the model reference trajectory as a desired behave.

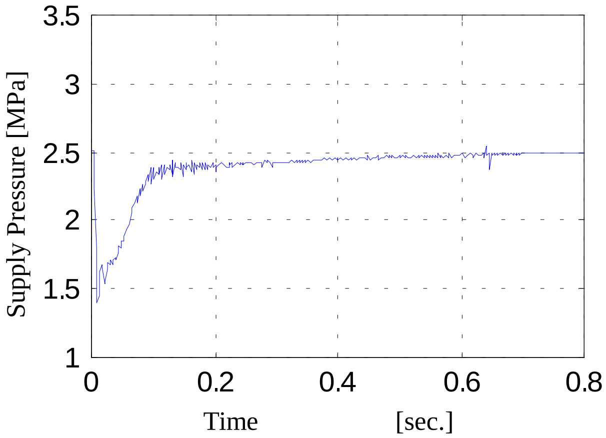


Figure 6 An axis supply pressure change with step reference input based on PID controller (all axes are on).

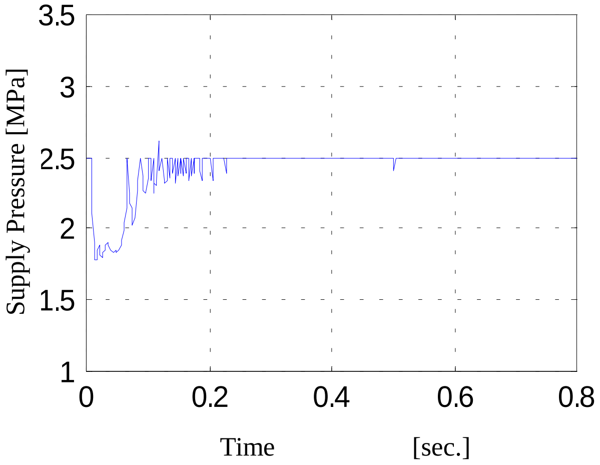


Figure 7 An axis supply pressure change with step reference input based on FMRLC (all axes are on).

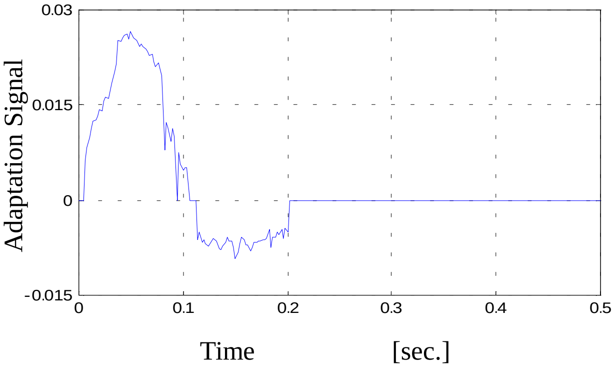


Figure 8 FMRLC adaptation signal with step reference.

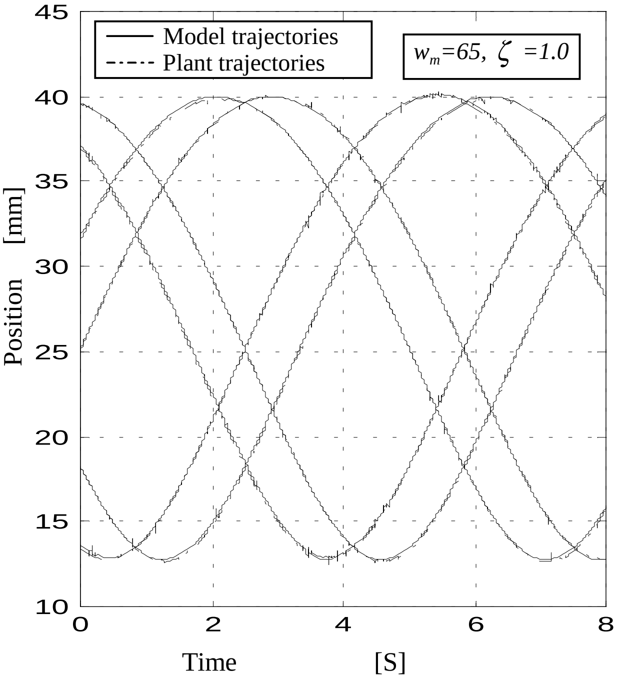


Figure 9 The axes trajectory response with circle reference of the TCP based on FMRLC.

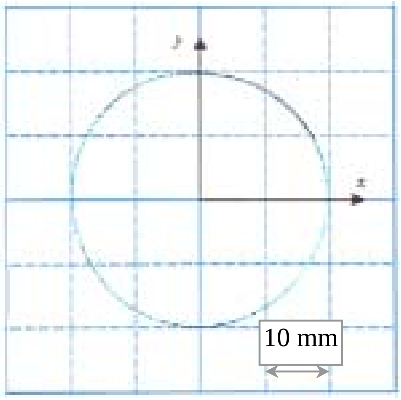


Figure 10 *x-y* record of the TCP with circle reference input based on FMRLC.

CONCLUSION

The control problem of an electro-hydraulic six axes system is studied in this paper. A self-learning adaptive controller based on fuzzy logic and control knowledge has been used to control the position and the motion of a Stewart platform manipulator. It is shown that the adaptive fuzzy controller has a good learning effect and a robust control performance against the nonlinearities and uncertainty of the system parameters. The controller does not need an off line training or a precise model of the plant. Through the experiments using the servo valves controlled system, the effectiveness and the robustness of the proposed control were confirmed.

The using of the proposed controller is limited by setting the model parameters i.e. for slower model the robustness continuous but for faster model the plant response will not be coincident with the model response and time lag will appear according to the limitation of the system speed.

Future research will involve a theoretical analysis for this method. Also, extending this policy for including the force control loops and calibration of the manipulator are the next targets of our research.

REFERENCES

[1] D. Li and S.E. Salcudean, Modeling, Simulation, and Control of a Hydraulic Stewart Platform, Proceeding of the IEEE international conference on Robotics and Automation, April 1997, pp. 3360-3366.

[2] G. Lebret, k. Liu and F. Lewis, Dynamics Analysis and Control of a Stewart platform manipulator, Journal of Robotic Systems, 10(5), 1993, pp. 629-655.

[3] H. Pang and M. Shahinpoor, Inverse Dynamics of a parallel Manipulator, Journal of Robotic Systems, 11(8), 1994, pp. 693-702.

[4] T. Chai and S. Tong, Fuzzy direct adaptive Control For A Class of Nonlinear Systems, Fuzzy set and systems, Vol. 103, 1999, pp. 379-387.

[5] K. Hayashi and A. Otsubo, Simulator For Studies of Fuzzy Control Methods, Fuzzy set and systems, Vol. 93, 1998, pp. 137-144.

[6] C. Li and R. Priemer, Fuzzy Control of Unknown Multiple Input Multiple Output Plants, Fuzzy set and systems, Vol. 95, 1998, pp. 295-306.

[7] J. R. Layne and K.M. Passino, Fuzzy Model Reference Learning Control for Cargo Ship Steering, IEEE control system Magazine, December 1993, 13(6), pp 23-34.

[8] J. R. Layne and K.M. Passino, Fuzzy Model Reference Learning Control for Cargo Ship Steering, Journal of Intelligent and fuzzy Systems, 1996, 4(1), pp 33-47.