FUZZY LOGIC CONTROL FOR DIRECT DRIVE ROBOT MANIPULATORS

°Chul-Goo Kang, Hee-Sung Kwak
Depart. of Mechanical Engineering, Kon-Kuk Univ.
Seong-dong-gu, Seoul 133-701, Korea

Abstract

This paper investigates the feasibility of applying fuzzy logic controllers to the motion tracking control of a direct drive robot manipulator to deal with highly nonlinear and time-varying dynamics associated with robot motion. A fuzzy logic controller with narrow shape of membership functions near zero and wide shape far away zero is analyzed. Simulation and experimental studies have been conducted for a 2 degree of freedom direct drive SCARA robot to evaluate control performances. Fuzzy logic controllers have shown control performances that are often better, or at least, as good as those of conventional PID controllers. Furthermore, the control performance of fuzzy logic controllers can be improved by selecting membership functions of narrow shapes near zero and wide shapes far away zero.

1. INTRODUCTION

In recent years, considerable attention has been given to the control of robotic manipulators with the increasing number of industrial robots in industry [5]. Conventional control method is first to derive the dynamic models of plants and then to design controllers based on the models [5]. However, the dynamics of direct drive robotic manipulators is highly nonlinear and time-varying [1], so the motion control problem of such manipulators is challenging. The control by way of the exact cancellation of all dynamic disturbances such as varying inertia, Coriolis and centrifugal accelerations, and gravitational and frictional disturbances, or the control by way of dealing with all dynamic disturbances as uncertainties, could be very complex and sometimes may not be computationally feasible for fast operating direct drive robots.

Another approach of the motion control of such manipulators is to use fuzzy logic control algorithms that are similar to human decision making procedures. Even though the dynamics of plants is very complex, the fuzzy logic controller (FLC) can be easily designed based on the heuristics about plant behaviors.

Fuzzy set theory is first introduced by L.A. Zadeh [14] to deal with imprecise objects in 1965. He suggested various possible application fields of fuzzy set theory (by relying on the use of linguistic variables and fuzzy algorithms) where the behaviors of systems are too complex or too ill-defined to admit of precise mathematical analysis [15]. In 1974, E.H. Mamdani applied successfully fuzzy logic algorithms for the control of a small laboratory steam engine [8]. After then, many applications of fuzzy logic control are reported in the various engineering fields including chemical processes and consumer products [7, 11, 12]. The results of these studies show that the performance of fuzzy logic controllers for slow plants or processes is better than or at least, as good as the one of conventional proportional -plus-integral-plus-derivative (PID) controllers. A good survey of the fuzzy logic controller until 1990 is presented by Lee [4]. Wang [13] viewed fuzzy control theory as a subset of nonlinear control theory. He pointed out the better control performance of the fuzzy controller compared to conventional PID controllers is due to the nonlinear behavior of the controller. Note that a PID controller is linear. Furthermore, he pointed out that, from a conceptual point of view, fuzzy controllers may indeed be robust because they are constructed from heuristics and human expertise, not from mathematical models, and the inaccuracy in the models should have less influence on the controller.

Motion control problems require faster and more accurate response compared with other industrial processes. Li and Lau [6] applied fuzzy logic control to servomotor systems and showed that the performance of fuzzy controller is better than that of PI controllers in terms of steady state error, settling time and response time through simulation study. In this study, they used two look-up tables for course and fine controls to cover the wide range of different situations the system may encounter. Huang and Tomizuka [2] applied fuzzy controller to two-dimensional motion tracking control and showed tracking precision and travel time can be improved compared with the system with conventional PD controller by a simulation study. In this study, they used monotonic membership functions for linguistic terms in the consequent of control rules. Scharf and Mandic [10] applied fuzzy controller to the motion control of an indirect drive robot manipulator and showed that the step response and tracking performance of the fuzzy controller is often superior to those of a conventional PID controller. However, the effect of time-varying and nonlinear dynamics in indirect drive manipulator with reduction gear is less than the one in direct drive manipulator, and the motion control problem is not so severe compared to direct drive manipulators.

In this paper, fuzzy logic control algorithms are applied to the motion control of direct drive robot manipulators. Instead of switching between course and fine controls, a fuzzy logic controller with membership functions of different shapes is tried in which linguistic terms have narrow shapes of membership functions near zero, and wide shapes far from zero. The effect of the sharpness of membership functions on fuzzy logic controllers is studied. Control performances of fuzzy controllers are analyzed and compared to those of conventional PID controllers by simulation and experimental studies. The robot used in this research is the two axes direct drive SCARA robot manipulator system constructed at the Department of Mechanical Engineering, Konkuk University.

2. TWO AXIS DIRECT DRIVE SCARA ROBOT SYSTEM

The schematics of two axes direct drive SCARA robot system is shown in Fig.1. This robot is constructed by the Department of Mechanical Engineering of Kon-Kuk University. This robot is composed of two NSK (Nippon Seiko Corp.) Megatorque motors and Drivers, two duralumin links and a base frame, a multifunctional I/O board from National Instruments, two counter boards, and an IBM PC/386 compatible.

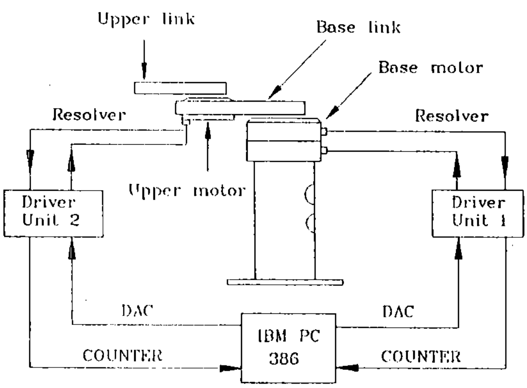

Fig.1. The schematics of two axes direct drive SCARA robot system.

The input voltages to the Drivers are -10 volts to 10 volts. The torque generated by the Megatorque motor is proportional to the input voltage. Maximum torques of the base motor and the upper motor are 147 N·m and 9.8 N·m, respectively. The resolver is attached to each Megatorque motor to measure the angular position and velocity. Resolver signal is converted to phase A and B quadrature signals by RDC (Resolver to Digital Converter) in the Driver Unit, and these quadrature signals are counted by each counter board made in the Department of Mechanical Engineering, Kon-Kuk University. The resolutions of feedback signals of the base motor and the upper motor are 38400 count/rev and 25600 count/rev, respectively. The maximum speeds of the base motor and the upper motor are 3 rev/sec and 4.5 rev/sec, respectively.

The dynamic model of the robot manipulator is not needed for designing a fuzzy logic controller, but we develop the dynamic model for a simulation purpose. The dynamics of n degree of freedom, rigid link, direct drive manipulators can be generally described as follows [3];

$$\mathbf{M}(\theta(t))\dot{\omega}(t)+\mathbf{v}(\theta(t),\omega(t))+\mathbf{g}(\theta(t))+\mathbf{f}(\omega(t),\tau(t))=\tau(t)$$

where θ is the joint angular position vector, ω is the joint angular velocity vector, τ is the input torque vector supplied by actuators, $\mathbf{M}$ is the symmetric and positive definite generalized inertia matrix, $\mathbf{v}$ is the vector due to Coriolis and centrifugal forces, $\mathbf{f}$ the friction torque vector, and $\mathbf{g}$ is the gravitational torque vector. For the two axes direct drive SCARA manipulator used in this paper, each term in the above equation is given as follows:

$$\theta(t)=\begin{bmatrix}\theta_1(t)\\ \theta_2(t)\end{bmatrix},\quad \omega(t)=\begin{bmatrix}\omega_1(t)\\ \omega_2(t)\end{bmatrix},\quad \tau(t)=\begin{bmatrix}\tau_1(t)\\ \tau_2(t)\end{bmatrix}$$

$$\mathbf{M}(\theta)=\begin{bmatrix}m_{11} & m_{12}\\ m_{12} & m_{22}\end{bmatrix},\quad \begin{aligned} m_{11}&=p_3+2p_1\cos\theta_2\\ m_{12}&=p_2+p_1\cos\theta_2\\ m_{22}&=p_2\end{aligned}$$

$$\mathbf{v}(\theta,\omega)=\begin{bmatrix}-\omega_2(2\omega_1+\omega_2)p_1\sin\theta_2\\ \omega_1^2 p_1\sin\theta_2\end{bmatrix}$$

$$\mathbf{f}(\omega,\tau)=\begin{bmatrix}f_1\\ f_2\end{bmatrix},\quad f_i=\begin{cases}T_i\operatorname{sign}\omega_i & \text{if } |\omega_i|>0\\ T_i\operatorname{sign}\tau_i & \text{if } \omega_i=0 \text{ and } |\tau_i|>T_i\\ \tau_i & \text{if } \omega_i=0 \text{ and } |\tau_i|\le T_i\end{cases}$$

$$T_1=3.2\ N\cdot m,\quad T_2=0.17\ N\cdot m$$

$$\mathbf{g}(\theta)=0$$

where

$$p_1=0.123+0.08M_p$$
$$p_2=0.138+0.0625M_p+I_p$$
$$p_3=1.676+0.165M_p+I_p$$
$$M_p=0 \text{ or } 3.76$$
$$I_p=0 \text{ or } 0.012$$

M_p is the mass of the payload and I_p is the mome payload. All the values above are represented in SI units.

3. DESIGN OF FUZZY LOGIC CONTROLLERS

Definitions and terminology

A *fuzzy set A* in X is characterized by a *membership function* $\mu(x)$ which associates with each point in X a real number in the interval [0, 1], with the value 0 representing non-membership and the value 1 representing full membership. The *support* of a fuzzy set A is the crisp set of all points x in X such that $\mu(x)>0$. A fuzzy set whose support is a single point in X with $\mu(x)=1$ is referred to as *fuzzy singleton*. Fuzzy set operations and fuzzy logic can be defined in many ways [9]. In the following, we summarize the definitions and terminology used in this paper. The membership function $\mu(x)$ of the *union* $A\cup B$ of two fuzzy sets A and B is pointwise defined for all x in X by

$$\mu_{A\cup B}(x)=\max\{\mu_A(x),\mu_B(x)\},\ x\in X$$

The membership function $\mu(x)$ of the intersection $A\cap B$ is pointwise defined for all x in X by

$$\mu_{A\cap B}(x)=\min\{\mu_A(x),\mu_B(x)\},\ x\in X.$$

The above definitions of the union and intersection of two fuzzy sets are originally proposed by Zadeh [14]. *Fuzzy implication* adopted in this paper is *Larsen's product rule* [4] among many others that is defined by

$$\mu_{A\to B}(x,y)=\mu_A(x)\mu_B(y),\ x\in X,\ y\in Y.$$

Note Mamdani proposed minimum operation $\min\{\mu_A(x),\mu_B(y)\}$ instead of the product $\mu_A(x)\mu_B(y)$ for the fuzzy implication [8]. These two operations are most frequently used for the fuzzy logic controller design. *Fuzzy reasoning* is generally based on a *compositional rule of inference*[15], which can be viewed as an extension of the *modus ponens*. The following *sup-min composition* proposed by Zadeh is used in this paper.

$$B'=A'\circ(A\to B)$$
$$\mu_{B'}(y)=\sup_{x\in X}\min\{\mu_{A'}(x),\mu_A(x)\mu_B(y)\}$$

A *linguistic variable* is a variable that can take words in natural languages (for example, big, small etc.) as its values. These words are usually labels of fuzzy sets.

Fuzzy logic controller

The block diagram of the overall fuzzy control system implemented in this paper is shown in Fig. 2. The sampler and zero-order hold (ZOH) are implemented by D/A converters (DAC). The position and velocity signals are digital, and fuzzy control actions are implemented by microprocessors and random access memories. The structure of the fuzzy logic controller is shown in Fig. 3.

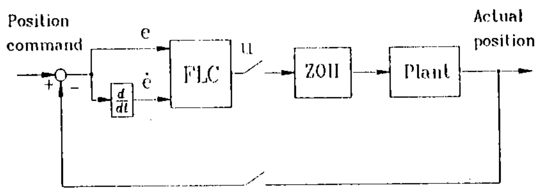

Fig.2. The block diagram of the overall fuzzy control system

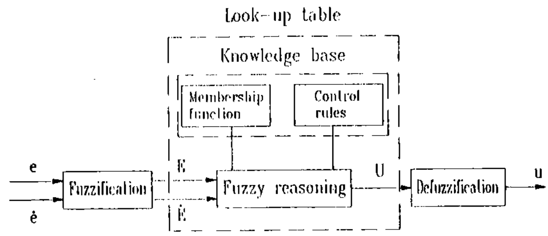

Fig. 3. The structure of the fuzzy logic controller

The input variables of the fuzzy logic controller are angular position error e (rad) and angular velocity error $\dot{e}$ (rad/s), and the output variable of the fuzzy logic controller is the control input u (N · m) to the motor driver, that is,

$$e=position\ command-actual\ position$$
$$\dot{e}=\frac{e(kT)-e((k-1)T)}{T}$$

where T represents a sampling time. Inside the fuzzy logic controller, $\dot{e}$, e and u are mapped to linguistic variables E, $\dot{E}$ and U by *fuzzification* operator, and the values of linguistic variables are composed of linguistic terms *PL* (Positive Large), *PM* (Positive Medium), *PS* (Positive Small), *ZO* (Zero), *NS* (Negative Small), *NM* (Negative Medium) and *NL* (Negative Large) which are all fuzzy sets. This linguistic term set forms a *fuzzy partition* of input and output spaces. The *knowledge base* of the fuzzy logic controller composed of a *data base* and a *rule base*. The data base defines membership functions for the above linguistic terms, and the rule base represents fuzzy control rules. Instead of quantizing the universe of discourse of input and output spaces into a finite number of segments (quantization levels), we standardize the position error, velocity error and control input values from -6 to 6. By doing in this way, we can keep infinite resolutions in the fuzzification and defuzzification operations. This method is a little different from the methods reported in the literature. Tab. 1 shows the standard values of the position error, velocity error and control input.

In the table, the maximum value of the control input is the same with the maximum torque the Megatorque motor can generate. The maximum values of the position and velocity error in the table correspond to scaling mapping. If these values become smaller, then tracking errors become smaller but the system generally becomes less stable. These values act like the inverse of the proportional and derivative gain in PD control. However, the difference of the FLC and PD controller is that the fuzzy logic controller acts in a nonlinear fashion.

Tab. 1. Standardization of the position error, velocity error and control input values.

Standard values	e_1, e_2 (*rad*)	$\dot{e}_1$, $\dot{e}_2$ (*rad/s*)	u_1 (*Nm*)	u_2 (*Nm*)
-6	≤-0.50	≤-1.00	-147.0	-9.80
-5	≤-0.41	≤-0.81	-122.5	-8.17
-4	≤-0.32	≤-0.64	-98.0	-6.53
-3	≤-0.23	≤-0.45	-73.5	-4.90
-2	≤-0.14	≤-0.27	-49.0	-3.27
-1	≤-0.05	≤-0.09	-24.5	-1.63
0	0.05≤ ≤0.05	0.09≤ ≤0.09	0.0	0.00
1	0.05≤	0.09≤	24.5	1.63
2	0.14≤	0.27≤	49.0	3.27
3	0.23≤	0.45≤	73.5	4.90
4	0.32≤	0.64≤	98.0	6.53
5	0.41≤	0.81≤	122.5	8.17
6	0.50≤	1.00≤	147.0	9.80

The membership functions of fuzzy sets (linguistic terms) have an effect on the control performance a lot. In this paper, two kinds of membership functions are used. One has same triangular shapes and same support sizes (Fig. 4), and the other has triangular shapes but different support sizes (Fig. 5). With the membership functions in Fig. 5, we emphasize that the control near zero is more important than the other portions.

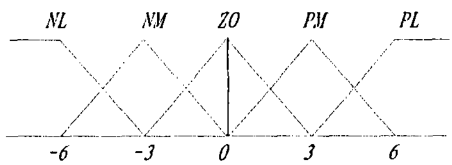

Fig. 4. Membership functions of fuzzy sets where support sizes are same (Case I).

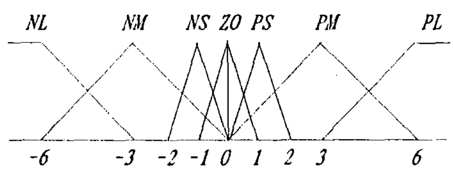

Fig. 5. Membership functions of fuzzy sets where support sizes are different (Case II).

A *fuzzy control rule* is a *fuzzy conditional statement* (IF-THEN statement) in which the antecedents and the consequent are associated with fuzzy concepts (linguistic terms). An example of a fuzzy control rule is as follows;

R_1: If E is *PS and* $\dot{E}$ is *NM*, then U is *PS*.

This rule is implemented by *fuzzy relation* R_1 = {(*PS* and *NM*) → *PS*} where '*and*' and '→' are processed by min operation (intersection) and Larsen's product rule, respectively. The whole fuzzy control rules can be represented symbolically by

$$R = also\ (R_1, R_2, \cdots, R_n).$$

The sentence connective '*also*' is operated by union operator. The whole fuzzy control rules for the fuzzy logic controller of the direct drive robot manipulator are shown in Tab. 2.

Tab. 2. The fuzzy control rules

E \ $\dot{E}$	NL	NM	NS	ZO	PS	PM	PL
NL	NL	NL	NL	NL	NL	NL	NL
NM	NL	NL	NL	NM	NM	NM	NL
NS	NL	NM	NM	NS	NS	NS	NM
ZO	NM	NS	NS	ZO	PS	PS	PM
PS	PM	PS	PS	PS	PM	PM	PL
PM	PL	PM	PM	PM	PL	PL	PL
PL	PL	PL	PL	PL	PL	PL	PL

Fuzzy reasoning (or approximate reasoning) is done by the sup-min composition operator. As an illustration, consider the following two rules.

R_1: if E is *PS and* $\dot{E}$ is *NM*, then U is *PS*,
R_2: if E is *PM and* $\dot{E}$ is *NS*, then U is *PM*.

Let the inputs be fuzzy singletons, namely, E' = 1 and $\dot{E}'$ = -1. Then the *inference mechanism* adopted in this fuzzy logic controller can be represented graphically as shown in Fig. 6.

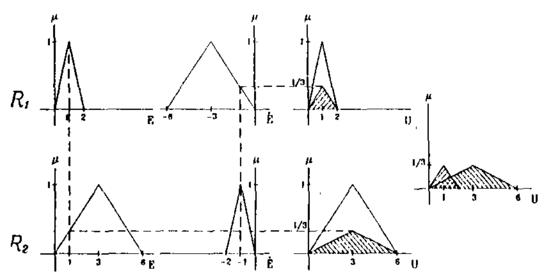

Fig. 6. Graphical representation of the inference mechanism

Defuzzification is a mapping from fuzzy control actions into nonfuzzy (crisp) control actions. Among many defuzzification strategies, we adopt *modified center average method* [13] defined by

$$U = \frac{\sum_i \overline{U}_i \times [\mu(\overline{U}_i) / \delta_i]}{\sum_i \mu(\overline{U}_i) / \delta_i}$$

where δ_i is a parameter characterizing the shape of a membership function such that the narrower the shape, the smaller is δ_i. This method is justified by our common sense that the sharper the shape of membership function, the stronger is our belief that the output U should be nearer to the center of the fuzzy set. As an illustration, the output U for the case of Fig. 6, is calculated as follows.

$$U = \frac{1\times[(1/3)/0.5]+3\times[(1/3)/1]}{(1/3)/0.5+(1/3)/1} = 1.667$$

The actual control inputs (N·m) are calculated by multiplying the scaling factors 147/6 (base motor) or 9.8/6 (upper motor). For given E' and $\dot{E}'$, we can calculate U each time as above, but this is time-consuming job. Therefore, we calculate U for given E' and $\dot{E}'$ in advance, and form a *look-up table* for the standard values of e and $\dot{e}$.

4. SIMULATION AND EXPERIMENTAL RESULTS

Digital computer simulation and experimental studies have been conducted for a 2 axis direct drive SCARA robot to evaluate the control performance for the fuzzy logic controller designed. In the simulation, the controller is realized as a discrete time model, and the manipulator as a continuous time model using a 4th order Runge-Kutta algorithm. The desired trajectory for both axes is given by a seventh order polynomial, simulating a pick-and-place job. The sampling time T is set to 5 msec in the simulation and experiment. The estimation of the calculation time of interrupt service routine within one sampling interval in the experiment was about 0.3 msec. Real-time program-ming with C using a timer interrupt function was done.

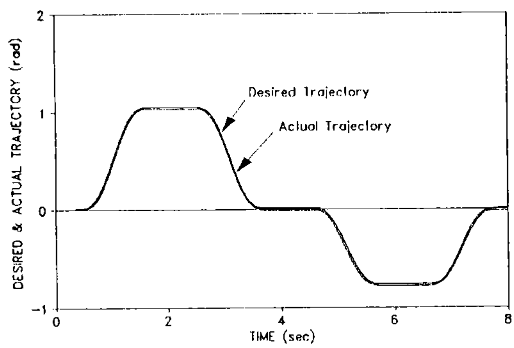

Fig. 7. Desired and actual position trajectories of the base axis in the simulation when the FLC (case II) is used without payload.

Fig. 7 shows the desired trajectory and actual trajectory of base axis in the simulation when a fuzzy logic controller (case II) is used without payload. Case I corresponds to the fuzzy logic controller using the same shapes of membership functions (see Fig. 4). Case II corresponds to the fuzzy logic controller using the different shapes (see Fig. 5). In Fig. 7, it's not easy to distinguish two trajectories, so in the following, we plot tracking errors in a bigger scale.

Fig. 8 and Fig. 9 show the position errors of the base and upper axis respectively in the simulation when a payload exists. For comparison, we plot the results of PID controller and fuzzy logic controllers of case I (Fuzzy 1) and case II (Fuzzy 2) together. Steady state error remained when the manipulator is stopped is due to Coulomb friction existing in the motor. In the PID controller we tried our best for PID gain tuning. From the figure, the fuzzy logic controller of case II is best in terms of position tracking errors. All the inputs in the simulations and experiments are within their limits. We don't plot the inputs due to the space limitation.

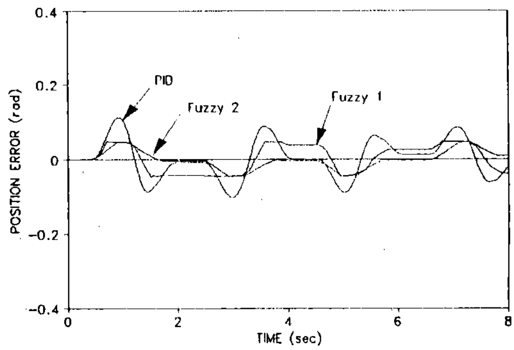

Fig. 8. The position tracking errors of the base axis in the simulation (payload = 3.76 kg)

Fig. 10 shows the position tracking errors of the upper axis in the experiment when the payload equals 3.76 kg. The behavior in the experiment is similar to the one in the simulation. The differences between the two (e.g., high frequency chattering) are believed to come from the unmodelled dynamics existing in the physical system.

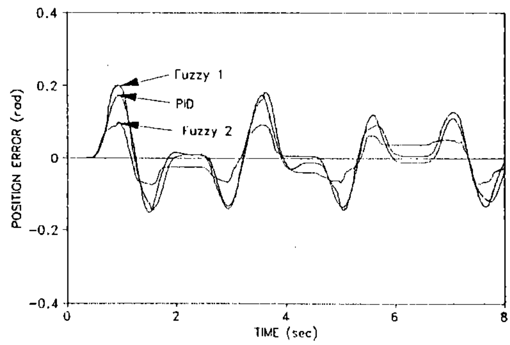

Fig. 9. The position tracking errors of the upper axis in the simulation (payload = 3.76 kg)

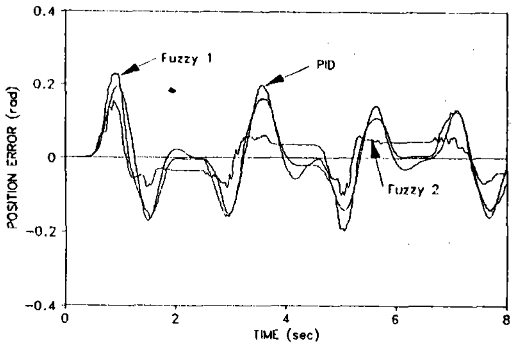

Fig. 10. The position tracking errors of the upper axis in the experiment (payload = 3.76 kg)

5. CONCLUSIONS

This work has demonstrating the feasibility of applying a fuzzy logic controller in the motion tracking control of a direct drive robot manipulator to deal with highly nonlinear and time-varying dynamics associated with robot motion. Fuzzy logic controllers have shown control performances that are often better, or at least, as good as that of conventional PID controller from the simulation and experimental studies. It is shown that the fuzzy logic controller with sharper membership functions near zero has better tracking performance but a little worse chattering. It is believed that these phenomena come from a gain-increasing effect by the sharper membership function near zero. One of the benefits of the fuzzy logic controller we experienced is its easiness to design and to understand. In the literature, several methods designing fuzzy logic controllers are proposed, and various applications of FLC are reported. However, the *analysis* of fuzzy logic control systems has not yet been well understood and that needs further research.

ACKNOWLEDGMENTS

This work was supported by Korea Science and Engineering Foundation under grant number 941-1000-066-2.

REFERENCES

[1] H. Asada and K. Youcef-Toumi, *Direct-Drive Robots: Theory and Practice*, The MIT Press, 1987

[2] L.-J. Huang and M. Tomizuka, "A self-paced fuzzy tracking controller for two-dimensional motion control," *IEEE Trans. Syst. Man Cybern.*, 1990, vol.20, no. 5, pp. 1115-1990

[3] C.G. Kang, R. Horowitz and G. Leitmann, "Robust deterministic control for robotic manipulators: Part I, II," *ASME DSC*, 1991, vol. 26, pp.171-186

[4] C.C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller-Part I, II," *IEEE Trans. Syst. Man Cybern.*, 1990, vol. 20, no. 2, pp. 404-435

[5] F.L. Lewis, C.T. Abdallah, *Control of Robot Manipulators*, Macmillan Publishing Co., 1993

[6] Y.F. Li and C.C. Lau, "Development of fuzzy algorithms for servo systems," *IEEE Control Systems Magazine*, April 1989, pp. 65-72

[7] J. Maiers and Y.S. Sherif, "Applications of fuzzy set theory," *IEEE Trans. Syst. Man Cybern.*, 1985, vol. SMC-15, no. 1, pp. 175-189

[8] E.H. Mamdani, "Application of fuzzy algorithms for simple dynamic plant," *Proc. IEE*, 1974, vol. 121, no. 12, pp. 1585-1588

[9] W. Pedrycz, *Fuzzy Control and Fuzzy Systems, second extended edition*, John Wiley & Sons, 1993

[10] E.M. Scharf and N. J. Mandic, "The application of a fuzzy controller to the control of a multi-degree-of-freedom robot arm," *Industrial Applications of Fuzzy Control*, M. Sugeno, ed., Elsevier Science Publishers B.V.(North-Holland), 1985, pp. 41-62

[11] M. Sugeno, "An introductory survey of fuzzy control, "*Infor. Sci.*, 1985, vol 36, pp. 59-83

[12] R.M. Tong, "A control engineering review of fuzzy systems," *Automatica*, 1977, vol. 13, pp. 559-569

[13] L.-X. Wang, *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*, Prentice-Hall, 1994, chapter 13

[14] L.A. Zadeh, "Fuzzy sets," *Informat. Control*, 1965, vol. 8, pp. 338-353

[15] L.A. Zadeh, "Outline of a new approach to the analysis complex systems and decision processes," *IEEE Trans. Syst. Man Cybern.*, 1973, vol. SMC-3, pp. 28-44