International Journal of Fuzzy Logic and Intelligent Systems, vol. 1, no. 1, June 2001 pp. 44-49

Design of Single-input Direct Adaptive Fuzzy Logic Controller Based on Stable Error Dynamics

Byung-Jae Choi

School of Computer and Communication Eng., Taegu University
15 Naeri Jinryang Kyungsan Kyungpook, 712-714, Korea

Abstract

For minimum phase systems, the conventional fuzzy logic controllers (FLC's) use the error and the change-of-error as fuzzy input variables. Then the control rule table is a skew symmetric type, that is, it has UNLP (Upper Negative and Lower Positive) or UPLN property. This property allowed to design a single-input FLC (SFLC) that has many advantages. But its control parameters are not automatically adjusted according to the situation of the controlled plant. That is, the adaptability is still deficient. We here design a single-input direct adaptive FLC (SDAFLC). In the AFLC, some parameters of the membership functions characterizing the linguistic terms of the fuzzy rules are adjusted by an adaptive law. The SDAFLC is designed by a stable error dynamics. We prove that its closed-loop system is globally stable in the sense that all signals involved are bounded and its tracking error converges to zero asymptotically. We perform computer simulations using a nonlinear plant and compare the control performance between the SFLC and the SDAFLC.

Key Words : Fuzzy Logic Control, Adaptive Control, Adaptation Law, Lyapunov Function. Hurwitz Polynomial

I. Introduction

Most conventional FLC's use the error and the change-of-error as fuzzy input variables regardless of the complexity of controlled plants. Either control input or incremental control input is typically used as the fuzzy output variable[1]. Such FLC's are suitable for simple lower order plants. In case of complex higher order plants, all process states are typically required. Then the design of a FLC is very difficult due to a number of tuning factors. Thus it is needed that a simple FLC that has quit a good control performance compared to those of the conventional FLC. One of those methods is the design of a SFLC.

Current controlled plants are more complex and large-scaled, and this tendency requires the development of more intelligent control schemes in the control field. To cope with such industrial tendency we design an AFLC equipped with an adaptation algorithm.

Many reports were introduced to simplify the design of the FLC's. Tang and Mulholland[2] showed that the FLC can be used as a multiband control. Li and Gatland[3] proposed a more systematic design method for PD and PI-type FLC's. Palm [4] proposed a sliding mode fuzzy controller. Choi et al.[5] designed the SFLC using a sole fuzzy input variable. However, in these cases there is no adaptation scheme. Some researches for AFLC's were performed by Wang[6-7]. He proposed various AFLC's such as direct and indirect AFLC's. In his AFLC's, excessive control actions could be derived in the supervisory control input to guarantee the stability of the closed-loop system.

In this paper, we first describe the design of a SFLC that uses a sole variable in the antecedent part of the fuzzy control rule, and then design a SDAFLC that adjusts some parameters of the SFLC directly. The SDAFLC is designed by using a stable error dynamics. We also perform computer simulations using a nonlinear plant and compare the control performance between the SFLC and the SDAFLC.

This paper is organized as follows. First the SFLC is simply described in Section II. A SDAFLC is designed by a stable error dynamics in Section III. Sections IV and V present computer simulations and concluding remarks, respectively.

II. Single-input FLC (SFLC)

Let the controlled process be a system with nth order (linear or nonlinear) state equation:

$$\begin{aligned} x^{(n)} &= f(\mathrm{x}, t) + b(\mathrm{x}, t)u(t) + d(t), \\ y &= x, \end{aligned} \qquad (1)$$

Maniscript received March 15, 2001; revised May 1, 2001.

with

$$\mathrm{x} = [x_1, x_2, \cdots, x_n]^T = [x, \dot{x}, \cdots, x^{(n-1)}]^T, \quad (2)$$

where $f(\mathrm{x}, t)$ and $b(\mathrm{x}, t)$ are partially known continuous functions, $d(t)$ is the unknown external disturbance, and $u(t) \in R$ and $y(t) \in R$ are the input and output of the system, respectively. $\mathrm{x}(t) \in R^n$ is the process state vector.

The control problem is to force $y(t)$ to follow a given bounded reference input signal $x_d(t)$. Let $e(t)$ be the tracking error vector as follows

$$\mathrm{e}(t) = \mathrm{x}(t) - \mathrm{x}_d(t) = [e, \dot{e}, \cdots, e^{(n-1)}]^T. \quad (3)$$

The rule form for the conventional (PD-type) FLC using two fuzzy input variables of the error and the change-of-error is as follows:

R^{ij}_{old}: If e is LE_i and $\dot{e}$ is LDE_j, then u is LU_{ij}

where $i = 1, 2, \cdots, M$, $j = 1, 2, \cdots, N$, and LE, LDE, and LU are the linguistic values taken by the process state variables e, $\dot{e}$, and u, respectively. Here the number of control rules is $M \times N$.

Table 1. Rule table for the conventional FLC.

$\dot{e}$ \ e	LE_{-2}	LE_{-1}	LE_0	LE_1	LE_2
LDE_2	LU_0	LU_{-1}	LU_{-1}	LU_{-2}	LU_{-2}
LDE_1	LU_1	LU_0	LU_{-1}	LU_{-1}	LU_{-2}
LDE_0	LU_1	LU_1	LU_0	LU_{-1}	LU_{-1}
LDE_{-1}	LU_2	LU_1	LU_1	LU_0	LU_{-1}
LDE_{-2}	LU_2	LU_2	LU_1	LU_1	LU_0

In Table 1, subscripts -2, -1, 0, 1, and 2 denote fuzzy linguistic values of Negative Big (NB), Negative Small (NS), ZeRo (ZR), Positive Small (PS), and Positive Big (PB), respectively.

Most rule tables for the minimum phase systems have a skew-symmetric property like Table 1. Note that the absolute magnitude of the control input $|u|$ is approximately proportional to the distance from the main diagonal line. If the quantization levels of the independent variables become infinitesimal, the boundaries of Table 1 become straight lines. Then the following switching line is introduced.

$$s_l: \quad \dot{e} + \lambda e = 0, \quad (4)$$

where $\lambda > 0$ is a slope of the switching line.

In turn a new variable called signed distance is derived as follows[5]:

$$d_s = \frac{\dot{e} + \lambda e}{\sqrt{1 + \lambda^2}}, \quad (5)$$

It represents the distance with a sign from $s_l = 0$ to an operating point. Then the following relation holds.

$$u \propto -d_s. \quad (6)$$

Now the control rule table can be established on an one-dimensional space of d_s instead of a two-dimensional space of the phase plane. Thus, the control action can be determined by only d_s, and the rule form is reduced as follows:

R^k_{new}: If d_s is LD_k then u is LU_k,

where LD_k is the linguistic value of the signed distance in the kth rule.

Above scheme can be extended to the case of a general n-input FLC. Similar to the two-dimensional rule table, the n-dimensional one for R^k_{GO} also satisfies the skew-symmetric property and the absolute magnitude of the control input is proportional to the distance from its main diagonal hyperplane[1]. Then the switching line s_l is changed to the following switching hyperplane S_l.

$$S_l: e^{(n-1)} + \lambda_{n-1} e^{(n-2)} + \cdots \lambda_2 \dot{e} + \lambda_1 e = 0. \quad (7)$$

Also, d_s of Eq. (5) is changed to a general signed distance D_s as follows:

$$D_s = \frac{e^{(n-1)} + \lambda_{n-1} e^{(n-2)} + \cdots \lambda_2 \dot{e} + \lambda_1 e}{\sqrt{1 + \lambda^2_{n-1} + \cdots \lambda^2_2 + \lambda^2_1}}. \quad (8)$$

That is, D_s represents the signed distance from the operating point to the switching hyperplane of Eq. (7). Then the rule table can be established on an one-dimensional space like Table 2.

Table 2. Rule table for the SFLC.

D_s	LDL_{-2}	LDL_{-1}	LDL_0	LDL_1	LDL_2
u	LU_2	LU_1	LU_0	LU_{-1}	LU_{-2}

In Table 2, LDL_k represents the linguistic value of the general signed distance in the kth rule. We call it a

SFLC (Single-input FLC).

As a results, the SFLC can replace most conventional FLC's with skew-symmetric property in the control rule table, and provides a simpler method for design of FLC's while achieving the desired control performance.

III. Design of SDAFLC

An adaptive controller is one in which some control parameters are updated by a learning rule. It can improve the overall control performance in the presence of large uncertainties or unknown variations in plant dynamics. An AFLC is a fuzzy logic system equipped with an adaptation algorithm. It has some advantages compared to conventional adaptive controllers. It is capable of incorporating linguistic fuzzy information from experienced human operators into a closed-loop control system.

Consider a fuzzy basis function (FBF). For this purpose, we rewrite a general fuzzy control rule for simplicity of notations as follows:

$$R^k: \text{ IF } x_1 \text{ is } A_k^1 \text{ and } \cdots \text{ and } x_n \text{ is} A_k^n, \text{ THEN } u \text{ is } B_k,$$

where $x = [x_1, x_2, \cdots, x_n]^T \subset U$ and $u \in R$. A_k^i and B_k are labels or linguistic values of fuzzy sets in U_i and R, respectively, and $k = 1, 2, \cdots, n_1$. n_1 is the number of total fuzzy control rules(m^n).

Let A_x be a fuzzy set in U, then the following definitions are given as follows:

Definition 1. The fuzzy logic systems with the singleton fuzzifier, product inference, and the height defuzzifier are of the following form:

$$u(x) = \frac{\sum_{k=1}^{n_1} \bar{u}^k \left(\prod_{i=1}^{n} \mu_{A_k^i}(x_i) \right)}{\sum_{k=1}^{n_1} \left(\prod_{i=1}^{n} \mu_{A_k^i}(x_i) \right)}, \qquad (9)$$

where $\bar{u}^k$ is the point in R at which μ_{B_k} achieves its maximum value. We assume that $\mu_{B_k}(\bar{u}^k) = 1$.

Definition 2. Define the fuzzy basis function as

$$\xi^k(x) = \frac{\prod_{i=1}^{n} \mu_{A_k^i}(x_i)}{\sum_{k=1}^{n_1} \left(\prod_{i=1}^{n} \mu_{A_k^i}(x_i) \right)}. \qquad (10)$$

Then a fuzzy logic system (9) can be rewritten as Eq. (11).

$$u(x) = \Theta_u^T \Xi_u(x), \qquad (11)$$

where $\Theta_u = [\bar{u}^1, \bar{u}^2, \cdots, \bar{u}^{n_1}]^T$ is an adjustable parameter vector, and

$$\Xi_u(x) = [\xi^1(x), \xi^2(x), \cdots, \xi^{n_1}(x)]^T$$

is a regressive vector composed of FBF's of Eq. (10). It was proved in [11] that the fuzzy logic systems in the form of Eq. (9) or (11) are universal approximators. Thus, the fuzzy logic system (9) is qualified as building blocks of the AFLC for nonlinear systems. We also see from Eq. (9) that linguistic information from experienced human operators in the form of the fuzzy IF-THEN rules can directly be incorporated into the controllers.

Now we can design a SDAFLC that adjusts some control parameters using the scheme of the SFLC. As stated in Section 2, the SFLC has the following one-dimensional control rule form:

$$R_{GN}^k: IF \;\; D_s \text{ is } LDL_k \;\; THEN \;\; u \text{ is } LU_k .$$

Then Eq. (9) is expressed by

$$u(D_s) = \frac{\sum_{k=1}^{n_2} \bar{u}^k (\mu_{LDL^k}(D_s))}{\sum_{k=1}^{n_2} (\mu_{LDL^k}(D_s))}, \qquad (12)$$

where $\bar{u}^k$ is the point in R at which μ_{LU^k} achieves its maximum value (assume that $\mu_{LU^k}(\bar{u}^k) = 1$), and n_2 is the number of one-dimensional control rules.

Similarly, the FBF is also summarized as

$$\xi^k(D_s) = \frac{\mu_{LDL^k}(D_s)}{\sum_{k=1}^{n_2} (\mu_{LDL^k}(D_s))}. \qquad (13)$$

Therefore an one-dimensional fuzzy rule R_{GN}^k can be expressed as a rigorous mathematical formula:

$$u(D_s) = \Theta_u^T \Xi_u(D_s), \qquad (14)$$

where $\Theta_u = [\bar{u}^1, \bar{u}^2, \cdots, \bar{u}^{n_2}]^T$ is an adjustable parameter vector, and

$\Xi_u(D_s) = [\xi^1(D_s), \xi^2(D_s), \cdots, \xi^{n_2}(D_s)]^T$ is a regressive vector.

The control purpose is to determine a feedback control input

$$u = u_f(D_s|\Theta_u) + u_a \qquad (15)$$

such that the tracking error should be as small as possible under some constraints, where u_f is a

SDAFLC and u_a is an auxiliary control input to ensure the closed-loop stability.

A SDAFLC u_f is designed by a Hurwitz error dynamics as follows.

Let $c = [c_n, c_{n-1}, \cdots, c_1]^T \in R^n$ be real valued vector such that all roots of the polynomial $h(s) = s^n + c_1 s^{n-1} + \cdots + c_n$ are in the open left-half plane, where s is the Laplace variable. If the functions f, b, and d are known in the controlled plant (1), then the control law is as follows:

$$u^* = b^{-1}(-f - d + x_d^{(n)} - c^T e), \quad (16)$$

where e is the tracking error vector that is given by Eq. (3). Substituting Eq. (16) into Eq. (1), the following error dynamics is obtained.

$$e^{(n)} + c_1 e^{(n-1)} + \cdots + c_n e = 0. \quad (17)$$

Since Eq. (17) is a Hurwitz from the definition of the constant parameter vector c, $\lim_{t\to\infty} e(t) = 0$.

But we don't know exact information about the functions f, b, and d, except for the sign of b(x,t). Substituting Eq. (15) into Eq. (1) and adding and subtracting bu^* in the right hand side, Eq. (1) is summarized as follows:

$$\begin{aligned} x^{(n)} &= f + b(u_f + u_a) + d + bu^* - bu^* \\ &= f + b(u_f - u^*) + bu_a + d \\ &\quad + (-f - d + x_d^{(n)} - c^T e) \\ &= b(u_f - u^*) + bu_a + x_d^{(n)} - c^T e. \end{aligned} \quad (18)$$

It can also be rewritten as Eq. (19) or (20).

$$e^{(n)} = b(u_f - u^*) + bu_a - c^T e. \quad (19)$$

$$\dot{e} = Ce + B(u_f - u^*) + Bu_a, \quad (20)$$

where

$$C = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ & & \cdot & & \\ & & \cdot & & \\ & & \cdot & & \\ -c_n & -c_{n-1} & -c_{n-2} & \cdots & -c_1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ b \end{bmatrix}, \quad (21)$$

Now we define the optimal parameter, Θ_u^*, and the minimum approximation error related to the control input, ε_u, as follows:

$$\Theta_u^* = \arg\min_{\Theta_u} [\sup_x |u_f(D_s|\Theta_u) - u^*|] \quad (22)$$

and

$$\varepsilon_u = u_f^* - u^*, \quad (23)$$

where $u_f^* = u_f(D_s|\Theta_u^*)$. Furthermore ε_u will maintain a very small value due to the universal approximating property of the fuzzy logic system[11]. That is,

$$|\varepsilon_u| = |u_f^* - u^*| \le \varepsilon, \quad (24)$$

where $\varepsilon > 0$ is a small value. Then the error equation (20) can be rewritten as

$$\begin{aligned} \dot{e} &= Ce + B(u_f - u_f^*) + B\varepsilon_u + Bu_a \\ &= Ce + B\Phi_u^T \Xi_u + B\varepsilon_u + Bu_a. \end{aligned} \quad (25)$$

where $\Phi_u = \Theta_u - \Theta_u^*$ and Ξ_u is the FBF that is defined by the Definition 2.

Now we replace the $u_f(D_s|\Theta_u)$ by a fuzzy logic system (14) and develop an adaptive law to update the parameter vector Θ_u. It is obtained by the following theorem.

Theorem: Consider the control law (16) and the stable error dynamics (17). If we choose the auxiliary control input u_a as

$$u_a \le -sgn(e^T PB)|\varepsilon_u|, \quad (26)$$

then the proposed system is stable in the sense of the Lyapunov and the parameter adaptation law is given as

$$\dot{\Theta}_u = -sgn(b)\gamma e^T P_n \Xi_u \quad (27)$$

where P is a positive definite symmetric n×n matrix that satisfies the Lyapunov equation

$$C^T P + PC = -Q. \quad (28)$$

Here, Q is an arbitrary positive definite matrix. $\gamma > 0$ is a constant that determines a kind of learning rate, and P_n is the last column of P.

Proof: Consider the following Lyapunov function candidate

$$V = \frac{1}{2} e^T Pe + \frac{|b|}{2\gamma} \Phi_u^T \Phi_u \quad (29)$$

Then,

$$\begin{aligned} \dot{V} &= -\frac{1}{2} e^T Qe + e^T PB(\Phi_u^T \Xi_u + \varepsilon_u + u_a) + \frac{|b|}{\gamma} \Phi_u^T \dot{\Phi}_u \\ &= -\frac{1}{2} e^T Qe + \frac{|b|}{\gamma} \Phi_u^T (\dot{\Phi}_u + sgn(b)\gamma e^T P_n \Xi_u) + \\ &\quad e^T PB\varepsilon_u + e^T PBu_a. \end{aligned} \quad (30)$$

From Eq. (30) we can get the following parameter adaptation law:

$$\dot{\Phi}_u = -sgn(b)\gamma e^T P_n \Xi_u. \quad (31)$$

Since $\dot{\Phi}_u = \dot{\Theta}_u$, Eq. (31) is equivalent to the adaptation law (27). Also if we choose the auxiliary control input such that the given condition (26) is satisfied, then Eq. (30) is summarized as follows:

$$\dot{V} \leq -\frac{1}{2} e^{T} Q e. \qquad (32)$$

Thus, the proposed SDAFLC is stable in the sense of the Lyapunov. □

IV. Simulation Example

We reveal the performance of the proposed SDAFLC via computer simulations. We consider a tracking problem for the inverted pendulum system. Fig. 1 shows the plant composed of a pole and a cart. The cart moves on the rail tracks in horizontal direction. The control objective is to balance the pole starting from an arbitrary condition by supplying a suitable force to the cart. For simplicity, we do not consider the position of the cart. The plant dynamics is then expressed as:

$$\ddot{\theta} = \frac{g\sin\theta + a\cos\theta - \mu_p w^2 l\cos\theta\sin\theta}{l(4/3 - \mu_p \cos^2\theta)}, \qquad (33)$$

$$\mu_p = \frac{m_p}{m_p + m_c}, \qquad (34)$$

$$a = \frac{F}{m_p + m_c}, \qquad (35)$$

where g is an acceleration due to gravity (=9.8m/sec2), and F is the applied force. m_c(=1.0kg) and m_p(=0.1kg) are masses and l(=0.5m) is the pole length.

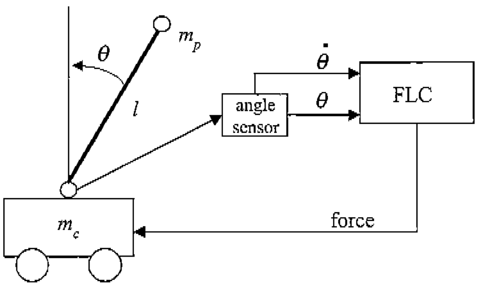


Fig. 1. The inverted pendulum system.

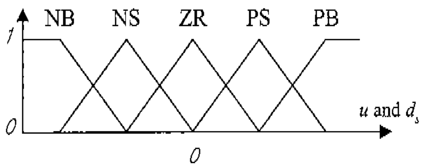
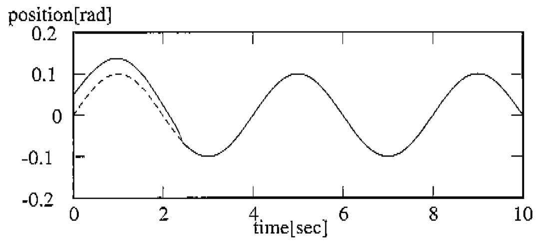


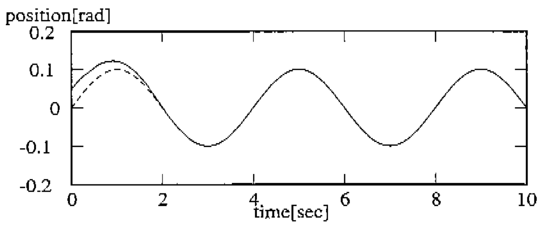
Fig. 2. The fuzzy sets for simulations.

Fig. 2 represents the fuzzy sets for control input and signed distance. In the SDAFLC, the center value of membership functions for the control input is automatically adjusted by an adaptation law. Simulation conditions of the SDAFLC are equally set to the case of the SFLC except for the addition of the parameter adaptation law. And we use the product inference and the height defuzzification.

Figures 3, 4, and 5 show the simulation results of tracking performances, control inputs, and tracking errors, respectively. Here (a) and (b) are the cases of the SFLC and the SDAFLC, respectively. Fig. 6 shows the response of the adaptive parameters of the SDAFLC. As shown in figures, the control performance of the SDAFLC is better than that of the SFLC. That is, an adaptive scheme can improve the performance of the conventional case.

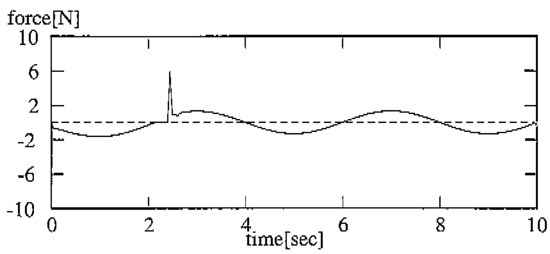


(a) SFLC.

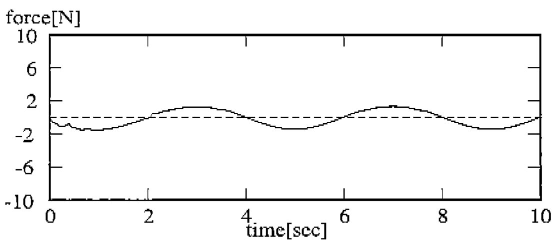


(b) SDAFLC.

Fig. 3. Comparison of tracking performances. (Solid line: Actual osition, Dashed line: Desired position)

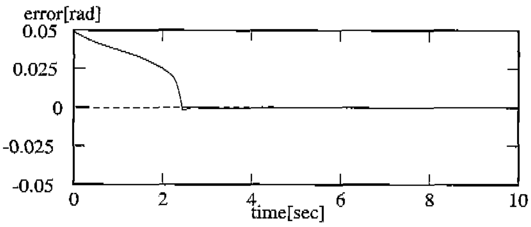


(a) SFLC.

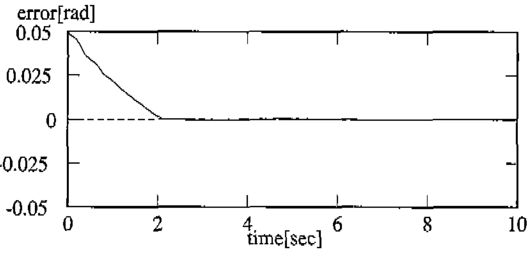


(b) SDAFLC.

Fig. 4. Comparison of control inputs.



(a) SFLC.



(b) SDAFLC.

Fig. 5. Comparison of tracking errors.

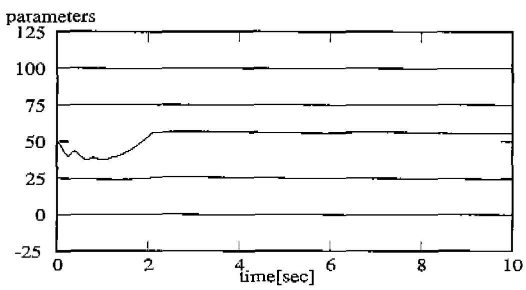


Fig. 6. Response of adaptive parameters.

V. Concluding Remarks

We first explained the SFLC and designed a SDAFLC. The SFLC was simply derived based on the property of the control rule table for conventional FLC's. This is extended to the general case of the nth order complex systems. Since the SFLC uses a sole fuzzy input variable of signed distance, it has many advantages: The number of fuzzy rules is greatly reduced compared to the case of conventional FLC's. In turn, generation, modification, and tuning of control rules are much easier.

We designed a SDAFLC using the concept of the SFLC. It was based on a stable error dynamics. And we ensured the closed-loop stability in the sense of the Lyapunov. Since the SDAFLC used a single fuzzy input variable, the FBF was considerably simplified. So the computational complexity was reduced. Furthermore we derived the auxiliary control input u_a that causes a very small control action differently from the case of [7]. Extremely, if we construct the rule base with sufficiently many rules, then the role of the auxiliary control input will disappear from the universal approximating property of a fuzzy logic system.

We also showed that an adaptive scheme can improve the control performance via computer simulations using the inverted pendulum system.

References

[1] D. Driankov, H. Hellendoorn, and M. Rainfrank, An Introduction to Fuzzy Control. Springer-Verlag Berlin Heidelberg, 1993.

[2] K. L. Tang and R. J. Mulholland, "Comparing fuzzy logic with classical controller design," *IEEE Trans. Syst.*, Man, Cybern., vol. 17, no. 6, pp. 1085-1087, 1987.

[3] H.-X. Li and H. B. Gatland, "Conventional Fuzzy Control and Its Enhancement," *IEEE Trans. Syst.*, Man, Cybern.(B), vol. 26, no. 5, pp. 791-797, 1996.

[4] R. Palm, "Robust Control by Fuzzy Sliding Mode", *Automatica*, vol. 30, no. 9, pp. 1429-1437, 1994.

[5] B.-J. Choi, S.-W. Kwak and B. K. Kim, "Design and Stability Analysis of Single-Input Fuzzy Logic Controller," *IEEE Trans. Syst.*, Man, Cybern.(B), vol. 30, no. 2, pp. 303-309, 2000.

[6] L. X. Wang, "Stable Adaptive Fuzzy Control of Nonlinear Systems," *IEEE Trans. Fuzzy Syst.*, vol. 1, no. 2, pp. 146-154, 1993.

[7] L.-X. Wang, Adaptive Fuzzy Systems and Control, Prentice-Hall International, Inc., New Jersey, 1994.

[8] H. K. Khalil, Nonlinear Systems, Macmillan Publishing Co., New York, 1996.

[9] J. J. Slotine and W. Li, Applied Nonlinear Control, Prentice-Hall, New Jersey, 1991.

Byung-Jae Choi

Byung-Jae Choi was born on August 20. 1964. He received the B.S. degree in Department of Electronic Engineering from Kyungpook National University, Korea, in 1987, and the M.S. and Ph. D. degrees in Department of Electrical Engineering from KAIST, Korea, in 1987 and 1998 respectively. Since 1999, he has been a faculty member of the School of Computer and Communication Engineering at the Taegu University, where he is currently an Assistant Professor. His research interests are Intelligent Control Theory and Applications. He is a member of IEEE, KFIS, KITE, KIEE, and ICASE.

Phone : +82-53-850-6633
Fax : +82-53-850-6619
E-mail : bjchoi@taegu.ac.kr