'95 KACC (1995. 10. 23 ~ 25)

ADAPTIVE FUZZY CONTROL BASED ON SPEED GRADIENT ALGORITHM

°Sacheul Jeoung, Byungkook Yoo, Woonchul Ham

Department of Electronic Engineer., Chonbuk National Univ., Chonju, Chonbuk, Korea 560-756
phone: +82-652-70-2405 ; fax: +82-652-70-2263 ; e-mail: woonchul@mips.chonbuk.ac.kr

Abstracts In this paper, the fuzzy approximator and nonlinear inversion control scheme are considered. An adaptive nonlinear control is proposed based on the speed gradient algorithms proposed by Fradkov. This proposed control scheme is that three types of adaptive law is utilized to approximate the unknown function f by fuzzy logic system in designing the nonlinear inversion controller for the nonlinear system. In order to reduce the approximation errors, the differences of nonlinear function and fuzzy approximator, another three types of adaptive law is also introduced and the stability of proposed control scheme are proven with SG algorithm.

Keywords Fuzzy Logic System, Adaptive Law, Speed Gradient Algorithm

1. INTRODUCTION

Since the original paper on fuzzy sets introduced by L.A.Zadeh in 1965, and the first fuzzy logic controller reported by Mamdani and his co-workers, several researches have used the concept of the fuzzy sets theory successfully applied to the area where the systems are complex and ill-defined.

For those systems whose accurate mathematical models are not avaiable or difficult to formulate, fuzzy control can often provide a good solution by incorporating linguistic informations from human experts. Despite its practical successes in many areas, fuzzy control seems to be deficient in formal analysis and robustness aspects. This is also a great resource of criticism from some conventional control researchers. To overcome this drawback, great efforts have been done in the field of fuzzy control during the recent years[3, 4, 5, 6, 7].

This paper is motivated by the researchers in Wang' works that fuzzy logic systems (center average defuzzifier, product-inference rule, singletone fuzzifier, and gaussian membership function) are capable of uniformly approximating any nonlinear function over compact input space[3]. In this paper, we introduce the speed gradient algorithm which was suggested by Fradkov based on the convexity and attainability. Three types of parameter update law are proposed based on the adaptive control scheme and error dynamics derived by Wang[3]. It is also demonstrated that the proposed adaptive control algorithm have the global stability. Especially, Wang utilized the general error dynamics of adaptive control to design the adaptive fuzzy controller. Many other researchers have attempted to apply the fuzzy approximator or fuzzy logic concepts to the conventional control technics. However, most of their works don't have formality. But in this work we use the SG algorithm which is formal to derive the update laws in the nonlinear fuzzy control. Also we derive another update laws which can compensate the approximate error based on SG algorithm.

This paper is organized as follows. Section 2 presents the general fuzzy logic system and fuzzy approximator. In section 3, the speed gradient algorithm which was suggestes by Fradkov based on the convexity and attainability are introduced[8]. In section 4, three types of parameter update law are suggested using fuzzy approximator which approximate the optimal nonlinear inversion control input based on speed gradient algorithm. Conclusions are drawn in the final section.

2. FUZZY LOGIC SYSTEM

2.1 Knowledge Base Constructed with Fuzzy Rules

The knowledge base for the fuzzy logic system comprises a collection of fuzzy IF-THEN rules. In this paper multiple-input single-out(MISO) rules will be used in the formulation of the control law. The MISO IF-THEN rule(s) are of the form

$$R^{(j)} \;:\; IF\ x_1\ is\ A_1^j\ and\ \cdots\ and\ x_n\ is\ A_n^j,\ THEN\ y\ is\ C^j \tag{2.1}$$

where $\underline{x} = (x, \cdots, x_n)^T \in V \subset R^n$ and $y \in W \subset R$ denote the linguistic variables associated with the inputs and output of the FLS. A_i^j and C^l are labels of the fuzzy sets in V and W, respectively, and i denotes the number of input(state) of FLS, *i.e.*, $i = 1, 2, \cdots, n$, and j denotes the number of rules of FLS, *i.e.*, $j = 1, 2, \cdots, M$. Fuzzy rule (2.1) can be implemented using fuzzy implication, which gives

$$A_i^j \times \cdots \times A_n^j \rightarrow C^j \tag{2.2}$$

which is a fuzzy set defined in the product space $V \times W$. Based on generalizations of implications in multivalue logic, many fuzzy implication rules have been proposed in the fuzzy logic literature. In this paper, we define the implication rule used t-norm operator, which gives

$$\mu_{A_1^j \times \cdots \times A_n^j \to C^j}(\underline{x}, y) = \mu_{A_i^j}(x_1) \star \cdots \star \mu_{A_n^j}(x_n) \star \mu_{C^j}(y) \quad (2.3)$$

where $\star$ denotes t-norm, which corresponds to the conjunction "min" or "product" in general.

2.2 Fuzzy Inference Engine

The fuzzy inference engine performs a mapping from fuzzy sets in V to fuzzy sets in R, based upon the fuzzy IF-THEN rules in fuzzy rule base and the compositional rule of inference.
Let B be a fuzzy set in V, then the fuzzy relational equation $B \circ R^j$, where "o" is the sup-star composition, results in M fuzzy stes. Using the t-norm operator yields

$$\mu_{B \circ R^j}(y) = sup_{\underline{x}}[\mu_B(\underline{x}) \star \mu_{A_1^j \times \cdots \times A_n^j \to C^j}(\underline{x}, y)] \quad (2.4)$$

In order to combine the M fuzzy sets into one fuzzy set t-norm can be employed, which results in

$$\mu_{B \circ (R^1, \cdots, R^{(M)})}(y) = \mu_{B \circ R^{(1)}}(y) \dot{+} \cdots \dot{+} \mu_{B \circ R^{(M)}}(y) \quad (2.5)$$

where $\dot{+}$ denotes the t-conorm(s-norm), the most commonly used operation for $\dot{+}$ is "max". If we use the product operation and choose $\star$ in (2.3) and (2.4) to be an algebraic product, then the inference is called product inference. Using product inference, (2.4) becomes

$$\mu_{B \circ R^j}(y) = sup_{\underline{x} \in V}[\mu_B(\underline{x}) \mu_{A_1^j}(x_1) \cdots \mu_{A_n^j}(x_n) \mu_{C^j}(y)]. \quad (2.6)$$

2.3 Fuzzifier

The fuzzifier maps a crisp point $\underline{x}$ into a fuzzy set B in V. In general, there are two possible choice of this mapping namely, singleton or nonsiglecton. In this paper, we use the sigleton fuzzifier mapping, *i.e.*,

$$\mu_B(\underline{x}') = \begin{cases} 1 & for \ \ (\underline{x}') = \underline{x} \\ 0 & for \ \ otherwise \end{cases}, \quad for \ \ \underline{x}' \in V. \quad (2.7)$$

2.4 Defuzzifier

The defuzzifier maps fuzzy sets in R to a crisp point in R. In general, there are three possible choices of this mapping namely, maximum, center-average, and modified center-average defuzzifier. In this paper, we use the center-average defuzzifier mapping, *i.e.*,

$$y = \frac{\sum_{j=1}^{M} \bar{y}^j (\mu_{B \circ R^{(j)}}(\bar{y}^j))}{\sum_{j=1}^{M} (\mu_{B \circ R^{(j)}}(\bar{y}^j))}. \quad (2.8)$$

where $\bar{y}^j$is the point in R at which μ_{C^j} achieves its maximum value (assume that $\mu_{C^j}(\bar{y}^j) = 1$).

2.5 Fuzzy Bases Function

The fuzzy logic system with *center-average defuzzifier* (2.8), *product inference* (2.6), and *singleton fuzzifier* (2.7) is of the following form:

$$y(\underline{x}) = \frac{\sum_{j=1}^{M} \bar{y}^j (\prod_{i=1}^{n} \mu_{A_i^j}(x_i))}{\sum_{j=1}^{M} (\prod_{i=1}^{n} \mu_{A_i^j}(x_i))} \quad (2.9)$$

If we fix the $\mu_{A_i^j}(x_i)$'s and view the $\bar{y}^j$'s as adjustable parameters, then (2.9) can be writen as

$$y(\underline{x}) = \theta^T \xi(\underline{x}) \quad (2.10)$$

where $\theta = (\bar{y}^1, \cdots, \bar{y}^M)^T$ is a parameter vector, and $\xi(\underline{x}) = (\xi^1(\underline{x}), \cdots, \xi^M(\underline{x}))^T$ is a regressive vector with the regressor $\xi^j(\underline{x})$ defined as

$$\xi(\underline{x}) = \frac{\prod_{i=1}^{n} \mu_{A_i^j}(x_i)}{\sum_{j=1}^{M} (\prod_{i=1}^{n} \mu_{A_i^j}(x_i))} \quad (2.11)$$

which are called *FBFs(fuzzy bases functions)*. These FBFs have been proved in [3] that they are universal approximators. We can fix all the parameters in $\xi^j(\underline{x})$ at the very beginning of the FBF expansion design procedure, so that the only free design parameters are θ_i. In this paper, we use this fuzzy logic system constructed FBFs with adaptive paramater vector θ as an alternative of unknown function u^* which is optimal control input in nonlinear inversion controller design under assumption that f is completely known. In this fuzzy modeling, however, we have to consider the error, the minimum approximation error. Therefore we have to find the update laws for the parameter vector θ and ω which grauntee the global stability. In later sections, we review three types of update law based on SG algorithm which can be found in formal and apply it to the adaptive fuzzy control scheme.

3. SPEED-GRANDIENT ALGORITHM

In this section, we shall introduce speed gradient algorithm and discuss three types of parameter update law with the stability analysis by using Lyapunov stability theory. Then we shall derive the new three types of parameter update laws based on the error dynamics derived in the previous section. First, we shall introduce the definition of convex function and the well known theorem concerning convex function briefly. Its contents play important role in the stability analysis of speed gradient algorithm.

Definition 3.1 *Let S be a convex set in R^n and let $f : S \to R^1$ be a real-valued function. We say that f* is a convex function on S *if and only if* $f[\lambda x_1 + (1-\lambda)x_2] \leq \lambda f(x_1) + (1-\lambda)f(x_2)$ *for all $x_1, x_2 \in S$ and for all λ such that* $0 \leq \lambda \leq 1$.

Note that convex functions are not defined if the domain is not a convex set.

Theorem 3.1 *Let S be a convex set in R^n and suppose that $f : S \to R^1$ is convex. Let x^0 be an interior point of S.*
(a) Then there are real numbers $a_1, a_2, \cdots, a_n$ such that

$$f(x) \geq f(x_0) + \sum_{i=1}^{n} a_i (x_i - x_i^0), \ \ where \ x \in S. \quad (3.1)$$

(b) If $f \in C^1$,i.e., first derivative of function f is continuous, on $S^{(0)}$, where $S^{(0)}$ denotes the set of interior points of S, then

$$a_i = \left.\frac{\partial f}{\partial x_i}\right|_{x=x^0} \cdot \qquad i = 1, \cdot, n. \qquad (3.2)$$

proof: See [11]

In general error dynamics of adaptive control system is a non-linear differential equation and can be expressed as

$$\dot{x}(t) = F(x, \phi, t), \qquad t \geq 0. \qquad (3.3)$$

Where $x(t) \in R^n$ is an error state vector, $\phi(t) \in R^m$ is a parameter estimation error vector ($\phi(t) = \hat{\theta}(t) - \theta^*$), $F(\cdot) : R^{n+m+1} \to R^n$ is a continuously differentiable vector function in x, θ . The control problem is to find the parameter update law

$$\hat{\theta}(t) = \Theta(x_0^t, \hat{\theta}_0^t, t), \qquad (3.4)$$

according to some criterion of "good" functioning of the system, where notation x_0^t and $\hat{\theta}_0^t$ mean the set { $x(s), 0 \leq s \leq t$ },{ $\hat{\theta}(s), 0 \leq s \leq t$ } respectively. Suppose this criterion requires to provide low values of some aim functional $Q_t = Q(x_0^t, \hat{\theta}_0^t, t)$. Typically $Q_t$may be local form such as $Q_t = Q(x(t), t)$, where $Q(x(t), t) \geq 0$ is a scalar smooth aim functional. Let us define a function $\tau(x, \hat{\theta}, t)$ as time derivative of Q_t (the speed of Q_t which changes along the trajectory of system). Then

$$\tau(x, \hat{\theta}, t) = (\nabla_x Q)^T F(x, \hat{\theta}, t) + \nabla_t Q \qquad (3.5)$$

where $\nabla_x Q$, and $\nabla_t Q$ denote the gradients of Q in x and t respectively.

With the above definition, we will introduce three types of parameter update law proposed by Fradkov.

Algorithm 3.1 .
(differential type)

$$\dot{\hat{\theta}}(t) = -\Gamma \nabla_{\hat{\theta}} \tau(x, \hat{\theta}, t) \qquad (3.6)$$

(integral type)

$$\hat{\theta}(t) = -\psi(x, \hat{\theta}, t) - \Gamma \int_0^t \nabla_{\hat{\theta}} \tau(x, \hat{\theta}, s) ds \qquad (3.7)$$

(finite type)

$$\hat{\theta}(t) = \theta^o(x, t) - \gamma(x, t)\psi(x, \hat{\theta}, t) \qquad (3.8)$$

where Γ is a symmetric, positive definite matrix, $\psi(\cdot)$ satisfies pseudo gradientity condition, *i.e.*, $\psi^T \nabla_{\hat{\theta}} \tau \geq 0$, where $\nabla_{\hat{\theta}} \tau$ denotes the gradient of τ in $\hat{\theta}$ and $\gamma(x, t) > 0$ is a scalar.

Theorem 3.2 *[8] Let system (3.3),(3.7) have unique solution for any initial conditions $x(0)$,$\hat{\theta}(0)$, and functions $F(x, \hat{\theta}, t)$, $\nabla_x Q(x, t)$, $\psi(x, t)$, $\nabla\tau(x, \hat{\theta}, t)$ be locally bounded in t (bounded in some region $\{(x, \hat{\theta}, t) : \|x\| + \|\hat{\theta}\| \leq \beta \leq \infty$, for $t \geq 0\}$) and following conditions beheld:*
(a) Growth condition: $\inf_t Q(x, t) \to \infty$ as $\|x\| \to \infty$.
(b) Convexity condition: function $\tau(x, \hat{\theta}, t)$ is convex in $\hat{\theta}$.
(c) Attainability condition: vector $\theta^* \in R^m$ and a function $\rho(Q)$ exists such that $\rho(Q) > 0$ when $Q > 0$ and

$$\tau(x, \theta^*, t) \leq -\rho(Q). \qquad (3.9)$$

Then all solutions of system (3.3),(3.7) are bounded and $Q_t \to 0$ as $t \to \infty$.

Proof: The proof is based on the Lyapunov-like function

$$V_t = Q_t + \frac{1}{2}(\hat{\theta}(t) - \theta^* + \psi(x, \hat{\theta}, t))^T \Gamma^{-1} (\hat{\theta}(t) - \theta^* + \psi(x, \hat{\theta}, t)) \qquad (3.10)$$

The time derivative of V_t along a trajectory of the system is given by

$$\dot{V}_t = \dot{Q}_t - (\hat{\theta}(t) - \theta^* + \psi(x, \hat{\theta}, t))^T \nabla_{\hat{\theta}} \tau(x, \hat{\theta}^*, t). \qquad (3.11)$$

From the pseudo gradientity condition $\psi^T \nabla_{\hat{\theta}} \tau \geq 0$, and convexity and attainability condition, the following inequalities can be drived:

$$\begin{aligned} \dot{V}_t &\leq \tau(x, \hat{\theta}, t) - (\hat{\theta}(t) - \theta^*)^T \nabla_{\hat{\theta}} \tau(x, \hat{\theta}^*, t) \\ &\leq \tau(x, \theta^*, t). \end{aligned} \qquad (3.12)$$

From (3.9),(3.12), $\dot{V}_t$ can be expressed as

$$\dot{V}_t \leq \tau(x, \theta^*, t) \leq -\rho(Q) < 0. \qquad (3.13)$$

Therefore $Q_t \to 0$ as $t \to \infty$. ∎

Theorem 3.3 *Let conditions of theorem 3.2 are fulfilled with $\rho(Q) \equiv 0$ in (3.9). Then all solutions of system (3.3),(3.6) are bounded.*

Proof: The proof is similar to Theorem 4.2 but in this case,

$$\dot{V}_t \leq \tau(x, \theta^*, t) \leq -\rho(Q) \leq 0. \qquad (3.14)$$

Therefore we can assure that Q_t is bounded. ∎

Theorem 3.4 *Let conditions of theorem 3.2 are fulfileed as well as strong pseudo gradientity condition*

$$\psi(x, t)^T \nabla_{\hat{\theta}} \tau(x, \hat{\theta}, t) \geq \kappa \|\nabla_{\hat{\theta}} \tau(x, \hat{\theta}, t)\|^\delta \qquad (3.15)$$

for some $\kappa > 0$ and $\delta \geq 1$ and inequality

$$\kappa\gamma(x, t)\|\nabla_{\hat{\theta}} \tau(x, \hat{\theta}, t)\|^{\delta-1} \geq \|\theta^o - \theta^*\| \qquad (3.16)$$

then all the solutions of system (3.3),(3.8) are bounded and $Q_t \to 0$ as $t \to \infty$.

Proof: Let $V_t = Q_t$. The time derivative of V_t along the system trjectory is given by

$$\dot{V}_t = \dot{Q}_t = \tau(x, \hat{\theta}, t) \qquad (3.17)$$

By convexity condition,

$$\begin{aligned} \dot{V}_t &= \tau(x, \hat{\theta}, t) \leq \tau(x, \theta^*, t) + (\hat{\theta} - \theta^*)^T \nabla_{\hat{\theta}} \tau \\ &= \tau(x, \theta^*, t) + (\hat{\theta} - \theta^o)^T \nabla_{\hat{\theta}} \tau + (\theta^o - \theta^*)^T \nabla_{\hat{\theta}} \tau \\ &= \tau(x, \theta^*, t) - \gamma(x, t)\psi(x, \hat{\theta}, t)^T \nabla_{\hat{\theta}} \tau \\ &+ (\theta^o - \theta^*)^T \nabla_{\hat{\theta}} \tau \end{aligned} \qquad (3.18)$$

From (3.15),

$$\begin{aligned} \dot{V}_t &\leq \tau(x, \theta^*, t) - \kappa\gamma(x, t)\|\nabla_{\hat{\theta}} \tau(x, \hat{\theta}, t)\|^\delta \\ &+ (\theta^o - \theta^*)^T \nabla_{\hat{\theta}} \tau \\ &\leq \tau(x, \theta^*, t) - \kappa\gamma(x, t)\|\nabla_{\hat{\theta}} \tau(x, \hat{\theta}, t)\|^\delta \\ &+ \|\theta^o - \theta^*\|\|\nabla_{\hat{\theta}} \tau\| \end{aligned} \qquad (3.19)$$

From (3.16),

$$\dot{V}_t \leq \tau(x, \theta^*, t) \leq -\rho(Q) < 0. \qquad (3.20)$$

Therefore $Q_t \to 0$ as $t \to \infty$. ∎

4. ADAPTIVE FUZZY CONTROL BASED ON SG ALGORITHM

In this section, we propose the three types of update law for nonlinear system which can be represented as a normal form. These update laws are developed by using fuzzy logic system and SG algorithm studied in previous section.

Consider the nth-order nonlinear systems of the form

$$x^{(n)} = f(\underline{x}) + bu, \tag{4.1}$$

where f is an unknown continuous function, b is a positive known constant, and $u \in R$ and $y \in R$ are the input and output of the system, respectively. We assume that the state vector $\underline{x} = (x_1, x_2, \cdots, x_n)^T = (x, \dot{x}, \cdots, x^{(n-1)})^T \in R^n$ is available for measurement. In the spirit of the nonlinear control literatures, these systems are in normal form and have the relative degree equal to n. The control objective is to force the state y to follow a given bounded reference signal $y_m(t)$, under the constraint that all signals involved must be bounded. More specifically, we now design adaptive fuzzy controller that achieve the following control objectives.

Control Objectives : Derive a feedback control $u = u(\underline{x}|\theta) + \hat{\omega}$ (bassed on fuzzy logic systems) and an speed gradient adaptive laws for adjusting the parameter vector θ such that the following two conditions are met :

1) The closed-loop system must be globally stable in the sense that all variables, $\underline{x}(t), \theta(t)$ and $u(\underline{x}|\theta)$, must be uniformly bounded; i.e., $|\underline{x}(t)| \leq M_x < \infty, |\theta(t)| \leq M_\theta < \infty$, and $|u(\underline{x}|\theta)| \leq M_u < \infty$ for all $t \geq 0$, where M_x, M_θ and M_u are design parameters specified by the designer.

2) The tracking error,$e \equiv y_m - y$, should be as small as possible under the constraints in 1). If the function f and the constant b are known, then by using the following

$$u^* = \frac{1}{b}(-f(\underline{x}) + y_m^{(n)} + \underline{k}^T\underline{e}). \tag{4.2}$$

We obtain error dynamics as follows from (4.1)

$$e^{(n)} + k_1 e^{(n-1)} + \cdots + k_n e = 0, \tag{4.3}$$

if we choose k appropriately, we can achive the quarantee that $\lim_{t\to\infty} e(t) = 0$. We define minimun approximation error ω as follows [3], *i.e.,*

$$\omega \equiv u_c(\underline{x}|\theta^*) - u^*, \tag{4.4}$$

and let $\hat{\omega}$ be a estimation of ω.

Generally f is unknown, the optimal control u^* cannot be implemented. Our purpose is to design a fuzzy logic system to approximate this optimal control.

Suppose that the control u is the summation of a fuzzy control $u_c(\underline{x}|\theta)$ and approximate compensation input $u_\omega = \hat{\omega}$:

$$u = u_c(\underline{x}|\theta) + \hat{\omega} \tag{4.5}$$

where $u_c(\underline{x}|\theta)$ is a fuzzy logic system in the form of (2.10) or (2.11). Substituting (4.4) into (4.1), we have

$$x^{(n)} = f(\underline{x}) + b(u_c(\underline{x}|\theta) + \hat{\omega}) \tag{4.6}$$

Now adding and substructing bu^* to (4.5) and after some straightforward manipulation, we obtain the error equation governing the closed-loop system:

$$e^{(n)} = -\underline{k}^T\underline{e} + b(u^* - u_c(\underline{x}|\theta) - \hat{\omega}) \tag{4.7}$$

or, equivalently,

$$\dot{\underline{e}} = \Lambda_c\underline{e} + \underline{b}_c(u^* - u_c(\underline{x}|\theta) - \hat{\omega}) \tag{4.8}$$

where

$$\Lambda_c = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 \\ & & & & \vdots & & \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 \\ -k_n & -k_{n-1} & & & \cdots & & -k_1 \end{bmatrix},$$
$$\underline{b}_c = (0, 0, \cdots, 0, b)^T. \tag{4.9}$$

In order to apply the speed gradient adaptive law, We choose the aim functional as follows

$$Q_t = \frac{1}{2}\underline{e}^T P\underline{e} \tag{4.10}$$

where P is positive definite symmetric matrix satisfying the Lyapunov equation

$\Lambda_c^T P + P\Lambda_c = -Q$, where $Q > 0$.

$$\tau = \dot{Q}_t = \underline{e}^T P(\Lambda_c\underline{e} + \underline{b}_c(-\omega + \phi^T\xi(\underline{x}) + \hat{\omega})) \tag{4.11}$$

where $\phi = \theta^* - \theta$. From the above equation, we can see that $\tau(\underline{x}, \theta, t)$ is linear in terms of $\theta, \hat{\omega}$ and that $\tau(\underline{x}, \theta, t)$ is convex function in $\theta, \hat{\omega}$. Now we choose $\psi_1(\underline{x}, \theta, t)$ and $\psi_2(\underline{x}, \hat{\omega}, t)$ defined in speed gradient algorithm as follow such that it satisfy pseudo gradient condition, *i.e.,* $\psi_1^T \nabla_\theta \tau \geq 0, \psi_2^T \nabla_{\hat{\omega}} \tau \geq 0$.

$$\psi_1(\underline{x}, \theta) = \nabla_\theta \tau = -\underline{e}^T P\underline{b}_c\xi(\underline{x}) \tag{4.12}$$
$$\psi_2(\underline{x}, \hat{\omega}) = \nabla_{\hat{\omega}}\tau = \underline{e}^T P\underline{b}_c \tag{4.13}$$

From the above discussion, we can see that all the conditions of theorem(growth condition, convexity condition and attainability condition) are satisfied. Therefore we can propose the following new three types of parameter update law for the adaptive control of nonlinear system which gaurantee the stability of the over all system.

Algorithm 4.2 .

(differential type)

$$\dot{\theta} = \Gamma\underline{e}^T P\underline{b}_c\xi(\underline{x}), \tag{4.14}$$
$$\dot{\hat{\omega}} = -\Gamma\underline{e}^T P\underline{b}_c. \tag{4.15}$$

(integral type)

$$\theta = -\psi_1(\underline{x}, \theta, t) + \Gamma\int_0^t \underline{e}^T P\underline{b}_c\xi(\underline{x})ds \tag{4.16}$$

$$\hat{\omega} = -\psi_2(\underline{x}, \hat{\omega}, t) - \Gamma\int_0^t \underline{e}^T P\underline{b}_c ds. \tag{4.17}$$

(finite type)

$$\theta(t) = \theta^0(x, t) - \gamma_1(x, t)\psi_1(x, \theta, t) \tag{4.18}$$
$$\hat{\omega}(t) = \hat{\omega}^0(x, t) - \gamma_2(x, t)\psi_2(x, \hat{\omega}, t) \tag{4.19}$$

5. CONCLUSION

In this paper, we developed an adaptive fuzzy controller 1) which does not require an accurate mathematical model of the system under control, 2) is capable of incorporating fuzzy control rules directly into the controllers, 3) gaurantees the global stability of the resulting closed-loop system in the sense that all signals involved are uniformely bounded, and 4) has three types of update laws which are formally obtained by the SG algorithm. And it is also shown that the parameter update law which was considered by Wang is a special type among the three types of parameter update law proposed in this paper. As a further study, we will look for a fuzzy modeling which can't be made as a normal form as a fuzzy liguistic model which is able to produce the error dynamics and then design the fuzzy adaptive controller based on SG algorithm.

REFERENCES

[1] C.C.Lee, " Fuzzy logic in control system : Fuzzy logic controller-Part I,II, " *IEEE Trans. on Systems, Man, and Cybernetics,* vol. SMC-20, no. 2, pp.404-435, 1990.

[2] H.J.Zimmermann, Fuzzy set theory and its applications, Kluwer Academic Pulishers, 1991.

[3] L.X.Wang, *Adaptive fuzzy systems and control*, Prentice-Hall International, Inc., 1994.

[4] M.B.Ghalia and A.T.Alouani, "Sliding mode control synthesis using fuzzy logic," *Proceedings of the American Control Conference*, Seattle , Washington, pp.1528-1532, June, 1995.

[5] G.C.Hwang and S.C.Lin, "A stability approach to fuzzy control design for nonlinear systems," *Fuzzy Sets and Systems*, vol.48, pp.279-287, 1992.

[6] R.Palm, "Sliding mode fuzzy control," *IEEE Int. Confer. on Fuzzy Systems*, San Diego, California, pp.519-526, 1992.

[7] A.T.Ollennu,B.A.Stacey and B.A.White "A direct adaptive fuzzy SMC," *IEEE Int. Conf. on Fuzzy Systems and Int. Fuzzy Engineering Symposium,* vol.4, pp.1979-1986, 1995.

[8] A.L.Fradkov, "Speed-gradient scheme and its application in adaptive control problems," *Automation and Remote Control*,pp.1342, 1979.

[9] W.C.Ham and J.J.Lee, "Adaptive nonlinear control of one-link flexible arm,"*IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, pp.3-7, Yokohama, Japan, 1993.

[10] W.C.Ham, "Adaptive control based on explicit model of robot manipulators," *IEEE trans. Automat. Contr.*, vol. 38, Apr, pp.654-658, 1993.

[11] W.C.Ham, "Adaptive Control Based on Speed Gradient Algorithm for Robot Manipulators," *International Conference on Industrial Electronics Control and Instrumentation*, vol. 2, pp.776-781, Bologna, Italy, 1994.