Adaptive Fuzzy Neural Control of Unknown Nonlinear Systems Based on Rapid Learning Algorithm

빠른 학습을 기반으로 한 미지의 비선형 시스템의 퍼지 신경망 적응 제어

김혜령,김재훈,김은태,박민용
연세대학교 전기 전자공학과

HyeRyeong Kim, Jae-Hun Kim, Euntai Kim, Mignon Park
DEPT. OF ELECTRICAL & ELECTRONIC ENG., YONSEI UNIVERSITY
E-mail : ynever@yonsei.ac.kr Tel: 02-2123-2868

Abstract - In this paper, an adaptive fuzzy neural control of unknown nonlinear systems based on the rapid learning algorithm is proposed for optimal parameterization. We combine the advantages of fuzzy control and neural network techniques to develop an adaptive fuzzy control system for updating nonlinear parameters of controller. The Fuzzy Neural Network(FNN), which is constructed by an equivalent four-layer connectionist network, is able to learn to control a process by updating the membership functions. The free parameters of the AFN controller are adjusted on-line according to the control law and adaptive law for the purpose of controlling the plant track a given trajectory and it 's initial values are off-line preprocessing. In order to improve the convergence of the learning process, we propose a rapid learning algorithm which combines the error back-propagation algorithm with Aitken's δ^2 algorithm. The heart of this approach is to reduce the computational burden during the FNN learning process and to improve convergence speed. The simulation results for nonlinear plant demonstrate the control effectiveness of the proposed system for optimal parameterization.

Key words : **Adaptive fuzzy neural control , Rapid learning , Aitken's δ^2, Optimal parameterization**

I. Introduction

Fuzzy controllers are supposed to work in situations where is a large uncertainty or unknown variation in plant parameters and structures. Generally, the basic objective of adaptive control is to maintain consistent performance of a system in the presence of these uncertainties. Therefore , advanced fuzzy control should be adaptive. An adaptive fuzzy system is a fuzzy logic system equipped with a training (adaptation) algorithm [1].

The most important advantage of adaptive fuzzy control over conventional adaptive control is that adaptive fuzzy controllers are capable of incorporating linguistic fuzzy information from human operators, whereas conventional adaptive controllers cannot. However, the fuzzy control techniques suffer from problems such as (1) the system performance relies significantly on so-called process experts who may not be able to transcribe their knowledge into the requisite rule form; (2) there exists no formal framework for the choice of the parameters of a fuzzy control system; [2]

To overcome the above-mentioned drawbacks, there is a growing interest in bringing the learning abilities of the neural networks to automate and realize the design of fuzzy control systems. The neural networks provide the connectionist structure (fault tolerance and distributed representation properties) and learning ability to the fuzzy logical systems.

In this paper , firstly , FNN are implemented using an equivalent four-layered connectionist network. With the result that FNN are able to find optimal initial parameters. Secondly, the proposed rapid learning algorithm which combines the error back-propagation algorithm with Aitken's δ^2 algorithm can reduce the computational burden during the FNN 's learning process and to improve convergence speed . The applicability and effectiveness of the proposed scheme were demonstrated through indirect adaptive fuzzy control system.

I. The Fuzzy Neural Networks and Learning Algorithm

1.1. The Fuzzy Neural Network structure

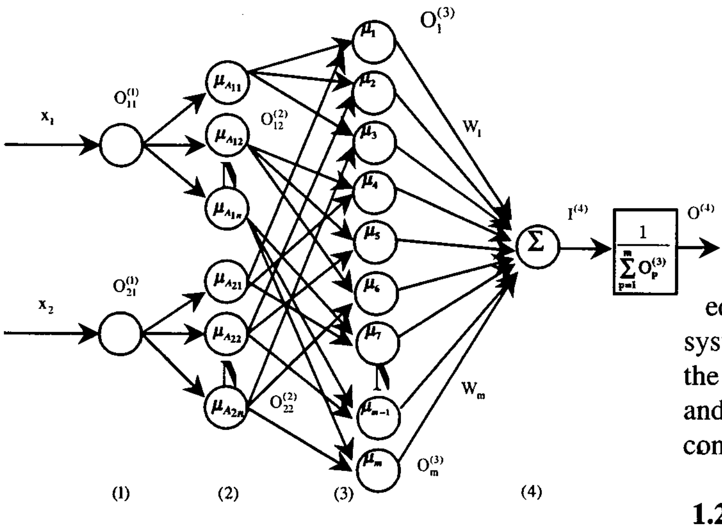

Fig.1 The structure of the Fuzzy Neural Network

Fig.1 depicts the FNN structure, which is a four-layer feed-forward connectionist network to realize fuzzy inference system. The FNN, in essence, integrates the basic elements and functions of a conventional Fuzzy Logic in a connectionist structure that has distributed learning ability to learn the membership functions and fuzzy logic rules. Let each input has n membership functions, then the input-output relations between layers are stated precisely as follows:

- *Layer 1: Input layer*

Input unit: $I_i^{(1)} = x_i, \quad i = 1,2$

Output unit: $O_{ij}^{(1)} = I_i^{(1)}, \quad i = 1,2 \ ; \ j = 1,2,\ldots,n$

- *Layer 2: Linguistic term layer*

Input units:

$$.I_{ij}^{(2)} = -\frac{(O_{ij}^{(1)} - a_{ij})^2}{b_{ij}^2}, \qquad i = 1,2; \ j = 1,2,\ldots,n$$

Output units:

$$O_{ij}^{(2)} = \mu_{A_{ij}} = \exp(I_{ij}^{(2)}), \qquad i = 1,2 \ ; \ j = 1,2,\ldots,n$$

where a_{ij} and b_{ij} are, respectively, the center and the width parameters of the Gaussian function.

- *Layer 3: Rule layer*

Input units: $I_{(j-1)n+l}^{(3)} = o_{1j}^{(2)} o_{2l}^{(2)}, \quad j = 1,2,\ldots,n;$
$l = 1,2,\ldots,n$

Output units:

$O_i^{(3)} = \mu_i = I_i^{(3)}, \quad i = 1,2,\ldots,m(= n^2)$

- *Layer 4: Output layer*

Input unit: $I^{(4)} = \sum_{p=1}^{m} O_p^{(3)} w_p$

Output unit: $O^{(4)} = \dfrac{I^{(4)}}{\sum_{p=1}^{m} O_p^{(3)}}$

Apparently, the FNN presented is equivalent to a simplified fuzzy inference system [3], where layers **1** and **2** correspond to the antecedent part of the fuzzy control rules, and the layers **3** and **4** correspond to the conclusion part.

1.2. A learning algorithm for the FNN

A. Gradient-based EBP algorithm.

Once an FNN has been constructed, learning aims at determining appropriate values for the parameters of the Gaussian MF, a_{ij} and b_{ij} and linking weights w_j .The adjustment of these parameters can be divided into two task, corresponding to IF (antecedent) part and THEN (consequence) part of the fuzzy logic rules.

Based on minimizing the error function of $E = (1/2)(y_d - y)^2$, the FNN parameters can be updated by

$$\begin{aligned} w_p(k+1) &= w_p(k) - \eta_p \frac{\partial E}{\partial w_p} \\ a_{ij}(k+1) &= a_{ij}(k) - \eta_a \frac{\partial E}{\partial a_{ij}} \quad i = 1,2 \\ b_{ij}(k+1) &= b_{ij}(k) - \eta_b \frac{\partial E}{\partial b_{ij}} \quad j = 1,2,\ldots,n \end{aligned} \qquad (1.1)$$

where η are the learning rate. In the updating rules, the gradients can be derived respectively by

$$\frac{\partial E}{\partial w_p} = -(y_d - y)\frac{O_j^{(3)}}{\sum_{p=1}^{m} O_p^{(3)}} \qquad (1.2)$$

$$\frac{\partial E}{\partial a_{1j}} = -(y_d - y)\frac{\partial y}{\partial o^{(4)}}\frac{2(\partial o_{1j}^{(1)} - a_{1j})o_{1j}^{(2)}}{b_{1j}^2(\sum_{p=1}^m o_p^{(3)})^2}\sum_{l=1}^n o_{2l}^{(2)} \qquad (1.3)$$

$$(w_{(j-1)n+l}\sum_{p=1}^m o_p^{(3)} - \sum_{p=1}^m o_p^{(3)} w_p), \quad j = 1, 2, \ldots, n$$

$$\frac{\partial E}{\partial a_{2j}} = -(y_d - y)\frac{\partial y}{\partial o^{(4)}}\frac{2(\partial o_{2j}^{(1)} - a_{2j})o_{2j}^{(2)}}{b_{2j}^2(\sum_{p=1}^m o_p^{(3)})^2}\sum_{l=1}^n o_{1l}^{(2)} \qquad (1.4)$$

$$(w_{(l-1)n+j}\sum_{p=1}^m o_p^{(3)} - \sum_{p=1}^m o_p^{(3)} w_p), \quad j = 1, 2, \ldots, n$$

$$\frac{\partial E}{\partial b_{1j}} = -(y_d - y)\frac{\partial y}{\partial o^{(4)}}\frac{2(\partial o_{1j}^{(1)} - a_{1j})^2 o_{1j}^{(2)}}{b_{1j}^3(\sum_{p=1}^m o_p^{(3)})^2}\sum_{l=1}^n o_{2l}^{(2)} \qquad (1.5)$$

$$(w_{(j-1)n+l}\sum_{p=1}^m o_p^{(3)} - \sum_{p=1}^m o_p^{(3)} w_p), \quad j = 1, 2, \ldots, n$$

$$\frac{\partial E}{\partial b_{2j}} = -(y_d - y)\frac{\partial y}{\partial o^{(4)}}\frac{2(\partial o_{2j}^{(1)} - a_{2j})^2 o_{2j}^{(2)}}{b_{2j}^3(\sum_{p=1}^m o_p^{(3)})^2}\sum_{l=1}^n o_{1l}^{(2)} \qquad (1.6)$$

$$(w_{(l-1)n+j}\sum_{p=1}^m o_p^{(3)} - \sum_{p=1}^m o_p^{(3)} w_p), \quad j = 1, 2, \ldots, n$$

B. Aitken 's δ^2 Learning Algorithm

In order to improve the convergence speed and release computational burden, we proposed a Rapid learning algorithm based on Aitken's δ^2 method [4]. Then the weight-updating formula is usually as sequence (1.1). Aitken's δ^2 is used to accelerate the learning speed.

The algorithm of the presented learning scheme is as follows.

<ALGORITHM>

Rapid learning method with Aitken's δ^2

```
Start learning with $w_0$;
For Learning does not finish ,
    $w_{n+2} = w_{n+1} - \eta \frac{\partial E}{\partial w_{n+1}}$;
    $w_n^* = w_n - \frac{(w_{n+1} - w_n)^2}{w_{n+2} - 2w_{n+1} + w_n}$; $\Leftarrow aitken's\ \delta^2$
    $w_n = w_n^*$ ;
End
```

The value w_n^* is a better approximation of the solution than w_n or w_{n+1}.

The rapid learning method not only accelerates the rate of convergence, but also induces convergence in some cases where the iteration diverges.

II. Description of Adaptive Fuzzy Systems

2. 1. Construct the Fuzzy Systems

The fuzzy inference engine uses the fuzzy IF-THEN rules to perform a mapping from an input linguistic vector $\mathbf{x}^T = [x_1, x_2, \mathrm{L}\ , x_n] \in R^n$ to and an output linguistic variable $y \in R$.

The i th fuzzy IF_THEN rule is written as

$$R^{(i)} : If\ x_1\ is\ A_1^i\ and\ \mathrm{L}\ \ and\ x_n\ is\ A_n^i \qquad (2.1)$$
$$Then\ y\ is\ B^i$$

where $A_1^i, A_2^i, \mathrm{L}\ , A_n^i$ and B^i are fuzzy sets [3]. Let M be the number of the fuzzy IF-THEN rules. By using product inference, center-average and singleton fuzzifier, the output of the fuzzy logic system can be expressed as

$$y(\mathbf{x}) = \frac{\sum_{i=1}^M \bar{y}^i(\prod_{j=1}^n \mu_{A_j^i}(x_j))}{\sum_{i=1}^M(\prod_{j=1}^n \mu_{A_j^i}(x_j))} = \boldsymbol{\theta}^T\xi(\mathbf{x}) \qquad (2.2)$$

$\bar{y}^i$	point at which $\mu_{B^i}(\bar{y}^i) = 1$;
$\boldsymbol{\theta}^T = [\bar{y}^1\ \bar{y}^2\ \mathrm{L}\ \bar{y}^m]$	adjustable parameter vector;
$\xi^T = [\xi^1\ \xi^2\ \mathrm{L}\ \xi^m]$	fuzzy basis vector ;
$\mu_{A_j^i}(x_j)$	membership function

where ξ^i is defined as

$$\xi^i(\mathbf{x}) = \frac{(\prod_{j=1}^n \mu_{A_j^i}(x_j))}{\sum_{i=1}^M(\prod_{j=1}^n \mu_{A_j^i}(x_j))} \qquad (2.3)$$

Then the output of the fuzzy system can be rewritten as (2.4)

$$\hat{f}(\mathbf{x}|\theta_f) = \frac{\sum_{l=1}^M \bar{y}_f^l [\prod_{i=1}^n \exp(-(\frac{x_i - \bar{x}_{fi}^l}{\sigma_{fi}^l})^2)]}{\sum_{l=1}^M [\prod_{i=1}^n \exp(-(\frac{x_i - \bar{x}_{fi}^l}{\sigma_{fi}^l})^2)]} = \boldsymbol{\theta}_f^T\xi(\mathbf{x})$$

2.2. Construct Adaptive Fuzzy Parameters

Step1. *FNN Learning based on Aitken's* δ^2

This is the same as the neural networks in the section I.

Step2. *Initial Membership parameters,*

$\bar{x}_{fi}^l, \sigma_{fi}^l$ *(off-line* $\xi^i(\mathbf{x})$ *'s initial parameter)*,

Construction

Step 3. *Construct the fuzzy system* $\hat{f}(\mathbf{x}|\theta f)$

Step 4. *On-line Adaptation*

III. Simulations

Consider the problem of balancing of an inverted pendulum on a cart[5] . Let $x_1 = \theta, x_2 = \theta\&$

$$\&_1 = x_2$$

$$\&_2 = \frac{g(x)\sin(x_1) - amlx_2^2(2x_1)/2 - a\cos(x_1)u}{4l/3 - aml\cos^2(x_1)} + d(t)$$

where $g = 9.8\,m/s^2$ the gravity constant, m is the mass of the pendulum($0.1kg$), M is the mass of the cart($M = 1kg$) , $2l$ is the length of the pendulum($1\,m$), $d(t)$ is the external disturbance and $a = 1/(m+M)$. Here, the reference signal is assumed as $y_r = (\pi/30)\sin(t)$.

The design parameters are selected $\gamma_1 = 50$ $p = \begin{bmatrix} 15 & 5 \\ 5 & 5 \end{bmatrix}, \gamma_2 = 1, Q = diag(10,10), \mathbf{k}_c = [1\ 2]^T$.

The following membership functions for x_i are given as

$$\mu^1_{A_i} = \exp\{-(x_i + 0.5)/0.13)^2\}, \mu^2_{A_i} = \exp\{-(x_i + 1)/0.13)^2\}$$

$$\mu^3_{A_i} = \exp\{-(x_i)/0.13)^2\}, \quad \mu^4_{A_i} = \exp\{-(x_i - 0.5)/0.13)^2\}$$

$$\mu^5_{A_i} = \exp\{-(x_i - 1)/0.13)^2\}, \quad i = 1, 2$$

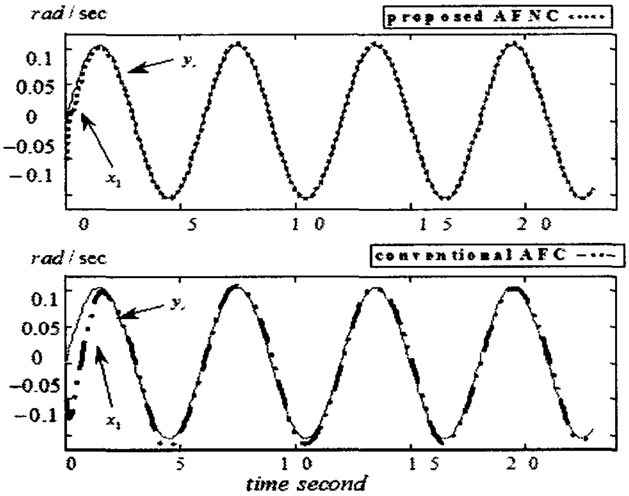

Fig.2 Trajectories of the x_1, y_r without disturbance

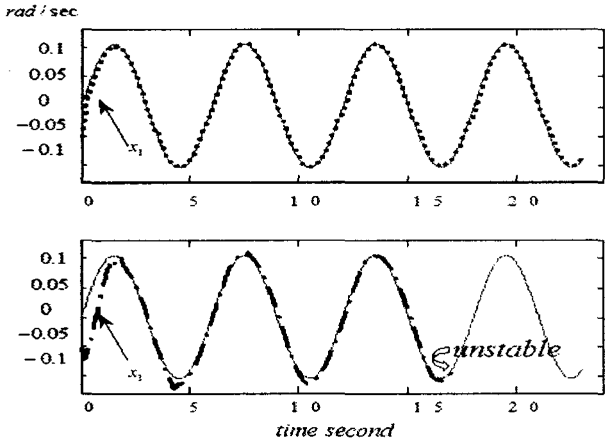

Fig.3 Trajectories of the x_1, y_r with disturbance $d(t)$

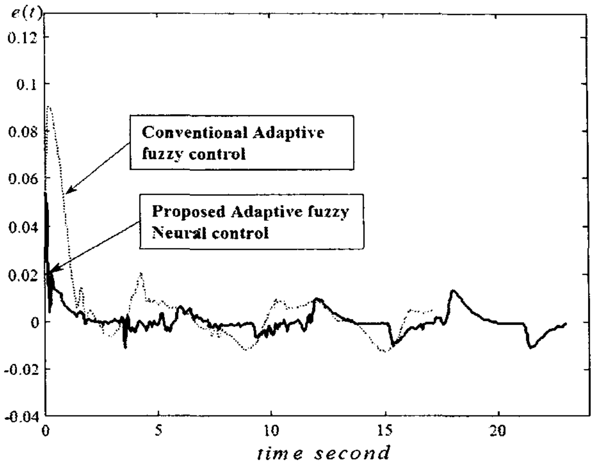

Fig.4 A comparison of the tracking output errors AFC and proposed AFNC based on Rapid algorithm

Fig.2 shows the rapid learning algorithm can reduce the computational burden and improve convergence speed.

Fig.3 shows adaptive fuzzy neural controller have better performance than conventional fuzzy control.

Fig.4 shows that the proposed fuzzy neural control system has a optimal performance and small tracking error.

IV. Conclusion

In this paper , the rapid learning algorithm based adaptive fuzzy neural controller of unknown nonlinear systems is developed . We propose Fuzzy Neural Network structure and learning with Aitken 's δ^2 algorithm. We can see that FNN are able to find optimal initial parameter selection during the learning

Through application examples, proposed system could give a good robust performance to the fuzzy control systems in the sense of an indirect adaptive control methodology.

REFERENCES

[1] L.X. Wang, Adaptive Fuzzy Systems and Control: Design and Stability Analysis. Englewood Cliffs, NJ: Prentice -Hall, 1994.

[2] G. stephanopoulos, C. Han, Intelligent systems in process engineering: a review, Computers Chem. Engng. 20,743-791,1996.

[3] J.E. Slotine and W. Li, Applied Nonlinear control, Englewood Cliffs, NJ: Prentice -Hall, 1991.

[4] A.C .Aitken,“ On Bernoulli's numerical solution algebraic equations” in Pros. Roy. Soc.Edinburgh, 289-305, 1926.

[5] L.X. Wang," Stable adaptive fuzzy controller with application to inverted pendulum tracking." IEEE Trans. Syt.,Man, Cybern., 26. 677-691 ,1996.