

## 퍼지신경망을 이용한 공 막대 시스템의 제어

강유원\*, 고재호\*, 류창완\*, 심재철\*, 배영철\*\*, 임화영\*

\*광운대학교 제어계측공학과 \*\*여수대학교 전기공학과

### Control of a Ball on Beam System using Fuzzy Neural Network

Kahng You-Won\*, Ko Jae-Ho\*, Ryu Chang-Wan\*, Shim Jae-Chul\*, Bae Young-Chul\*\*, Yim  
Wha-Yeong\*

**Abstract** - Neural Network has advantages of learning and normalizing capabilities. Fuzzy controller is based on a fuzzy logic that is so effective to represent uncertain phenomena of real world and make its approximation.

In this paper, Fuzzy Neural Network controller which equipped with adaptive control algorithm is described.

Proposed Fuzzy Neural Network Controller applied to a ball on beam system which have nonlinear characteristics shows a good performance.

타내면 그림 1과 같다.

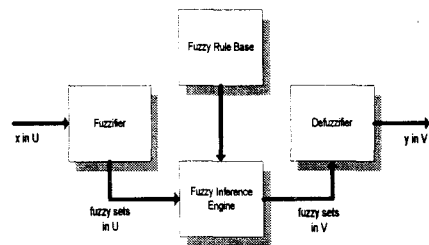


그림 1. 퍼지 시스템의 블록선도

Fig. 1. Block diagram of Fuzzy System

## 1. 서 론

신경망은 학습능력과 일반화 능력 등의 제어에 적합한 장점들을 가지고 있다. 퍼지제어는 제어대상체로부터 발생하는 일련의 동작들을 조절하기 위한 퍼지 규칙의 집합체로 구성되고, 실세계의 근사적이고 불확실한 현상을 기술하는데 효과적이거나 소속함수와 제어규칙을 설정하는데 많은 시행착오가 요구된다.

본 논문에서는 제어대상체로 비선형적 특성을 가진 공 막대 시스템을 정하고 막대에서의 공의 위치 제어를 위해 실세계에 대한 전문가적 경험과 지식을 표현할 수 있고 유연성과 강건성을 가진 퍼지제어와 학습과 일반화 능력을 가진 신경망의 장점들을 결합한 퍼지 신경망 제어 기법을 제시하여 소속함수를 설정한 후 오차 역전파 알고리즘을 통해 소속함수의 파라미터들을 갱신하여 나은 성능을 제시하고자 한다.

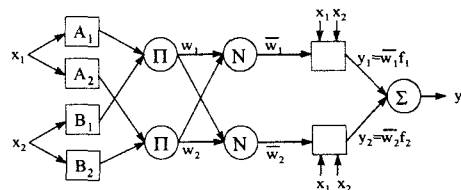
## 2. 본 론

### 2.1 퍼지 시스템

퍼지 시스템의 핵심적인 부분은 언어적 형태로 표현된 일련의 제어 규칙으로 전문가의 지식, 운전자의 경험 등을 그대로 적용할 수 있는 장점이 있다.

퍼지 시스템은 퍼지화기, 비퍼지화기, 퍼지 규칙 기반, 퍼지 추론 엔진으로 구성되며 블록선도로 나

### 2.2 퍼지 신경망



Layer1 Layer2 Layer3 Layer4 Layer5

그림2. 퍼지제어의 전방향 네트워크

Fig2. Feedforward Network of Fuzzy Control

위의 그림은 입력  $x_1, x_2$ 와 출력  $y$ 를 갖는 퍼지제어기의 구조이다.

신경망은 비선형 최적화 알고리즘의 하나로 복잡한 수식계산 과정이 필요없이 단순한 반복적 연산에 의해 수행되므로 비선형 문제에 쉽게 적용할 수 있다.

퍼지제어의 전방향 네트워크화를 하여 신경망의 역전파 알고리즘을 적용할 수 있다.

### 2.3 공 막대 시스템

공 막대 시스템을 그림 2에 나타냈으며,

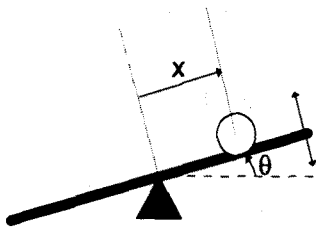


그림 3. 공 막대 시스템  
Fig3. Ball on Beam System

모델링은 라그랑지안을 이용하면 식(1)에서,

$$0 = \left(\frac{J_b}{R^2} + M\right) \ddot{r} - Mr \ddot{\theta}^2 + MG \sin \theta$$

$$u = (J + J_b + Mr^2) \ddot{\theta} + 2Mr \dot{r} \dot{\theta} + MG \cos \theta \quad (1)$$

상태 변수를 다음과 같이 정의한 후,

$$[x_1 \ x_2 \ x_3 \ x_4] = [r \ \dot{r} \ \theta \ \dot{\theta}]^T \quad (2)$$

정리하면 식(3)로 표현된다.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} x_2 \\ B(x_1 x_3^2 - G \sin x_3) \\ x_4 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u$$

$$y = x_1 \quad (3)$$

여기서, 파라미터는  $B=0.7143$ ,  $G=9.81$ 이다.

## 2.4 퍼지신경망의 구조

퍼지신경망 제어에서 입력변수는  $[x,y,z]=[x, \theta, \dot{x}]$ , 출력, 즉 모터에 인가되는 제어입력변수는  $[u]=[ \ddot{\theta} ]$ 이다.

**layer 1** 입력  $x, y, z$  에 대한 소속함수는  $\{a,b,c\}$ 의 조건부 파라미터를 갖는 종형(bell) 함수를 사용한다.

$$\mu(x) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (4)$$

**layer 2**  $\Pi$ , 곱(product)을 사용한다.

$$\omega = \mu_A(x) \mu_B(y) \mu_C(z) \quad (5)$$

**layer 3**  $N$ , 일반화(normalization)방법을 사용한다.

$$\bar{\omega}_i = \frac{\omega_i}{\omega_1 + \omega_2 + \dots + \omega_n} \quad (6)$$

**layer 4** 결론부 파라미터  $\{p,q,r,s\}$ 를 갖는 1차 Sugeno 퍼지 모델을 사용한다.

$$\text{if } x \text{ is } A \text{ and } y \text{ is } B \text{ and } z \text{ is } C \text{ then } f = p*x + q*y + r*z + s \quad (7)$$

**layer 5** 입력되는 신호들의 합으로 출력을 계산한다.

$$\text{output} = u = \sum \bar{\omega}_i f_i = \frac{\sum \omega_i f_i}{\sum \omega_i} \quad (8)$$

### 2.3.1 역전파 알고리즘

다층 신경망을 통해 얻고자 하는 목표 출력과 실제 출력과의 오차를 이용해서 오차를 줄이는 방향으로 각 층의 가중치를 갱신하는 방법을 사용한다. 조건부 파라미터  $\{a,b,c\}$ 와 결론부 파라미터  $\{p,q,r,s\}$ 값들을 변경하여 최소자승오차를 최소화한다.

### 2.3.1 최급강하법

오차측정

$$E = \frac{1}{2} \sum \|e\|^2 = \frac{1}{2} \sum \|x_3\|^2 \quad (9)$$

각도에 대한 오차측정을 하여 파라미터 변경을 한다.

$$\frac{\partial E}{\partial (a, b, c)} = \frac{\partial E}{\partial x_3} \frac{\partial x_3}{\partial u(k-1)} \frac{\partial u(k-1)}{\partial \omega_i} \frac{\partial \omega_i}{\partial \mu} \frac{\partial \mu}{\partial (a, b, c)}$$

$$\frac{\partial E}{\partial (p, q, r, s)} = \frac{\partial E}{\partial x_3} \frac{\partial x_3}{\partial u(k-1)} \frac{\partial u(k-1)}{\partial (p, q, r, s)} \quad (10)$$

$$\begin{bmatrix} a_{next} \\ b_{next} \\ c_{next} \end{bmatrix} = \begin{bmatrix} a_{now} \\ b_{now} \\ c_{now} \end{bmatrix} + \eta \begin{bmatrix} \Delta a \\ \Delta b \\ \Delta c \end{bmatrix} = \begin{bmatrix} a_{now} \\ b_{now} \\ c_{now} \end{bmatrix} - \eta \begin{bmatrix} \sum \frac{\partial E}{\partial a} \\ \sum \frac{\partial E}{\partial b} \\ \sum \frac{\partial E}{\partial c} \end{bmatrix} \quad (11)$$

$$\begin{bmatrix} p_{next} \\ q_{next} \\ r_{next} \\ s_{next} \end{bmatrix} = \begin{bmatrix} p_{now} \\ q_{now} \\ r_{now} \\ s_{now} \end{bmatrix} + \eta \begin{bmatrix} \Delta p \\ \Delta q \\ \Delta r \\ \Delta s \end{bmatrix} = \begin{bmatrix} p_{now} \\ q_{now} \\ r_{now} \\ s_{now} \end{bmatrix} - \eta \begin{bmatrix} \sum \frac{\partial E}{\partial p} \\ \sum \frac{\partial E}{\partial q} \\ \sum \frac{\partial E}{\partial r} \\ \sum \frac{\partial E}{\partial s} \end{bmatrix} \quad (12)$$

### 2.2.1 시뮬레이션

초기소속함수는 입력  $[x,y,z]=[x, \theta, \dot{x}]$ 에 대한 종형(bell) 함수이다.

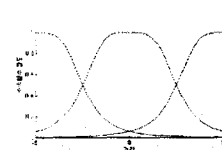


그림 4. 거리에 대한 소속함수

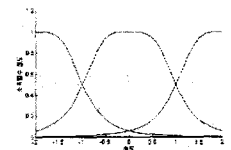


그림 5. 속도에 대한 소속함수

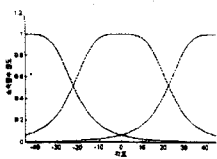


그림 6. 각도에 대한 소속함수

초기조건  $[x, \dot{x}, \theta, \dot{\theta}] = [0 \ 0 \ 5 \ 0]$  일 때의 출력이다.

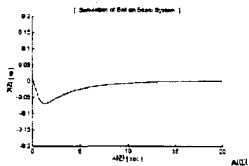


그림 7. 거리

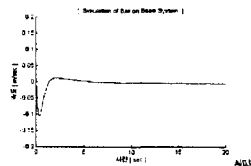


그림 8. 속도

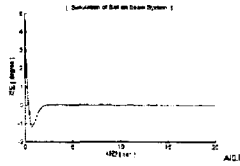


그림 9. 각도

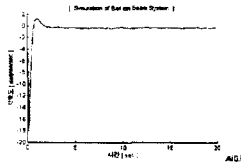


그림 10. 각속도

파라미터 변경 후의  $[x,y,z]=[ \dot{x}, \theta, \dot{x}]$ 에 대한 소속함수의 형태이다.

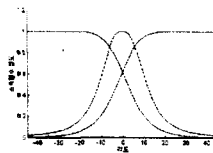


그림 11. 각도

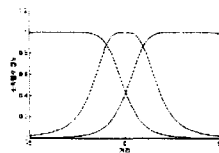


그림 12. 거리

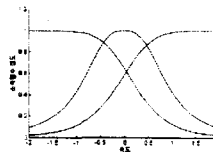


그림 13. 속도

### 3. 결 론

퍼지 제어기에 있어서 소속함수의 설정은 매우 중요하다. 본 논문에서 제안한 퍼지신경망 제어기는 소속함수의 파라미터 결정에 신경망의 역전파 알고리즘을 도입하였다. 최급강화법을 이용한 신경망의 학습을 통해 파라미터가 변화하여 제어 성능의 향상을 나타내었으며 제어 목표에 맞도록 수렴

하였다.

### [참 고 문 헌]

- [1] Li-Xin Wang, "A Course in Fuzzy Systems and Control", Prentice-Hall International, Inc, 1997
- [2] Li-Xin Wang, "Adaptive Fuzzy Systems and Control-Design and Stability Analysis", Prentice-Hall, 1994
- [3] Jang, J.-S., C.-T. Sun, and Mizutani, Neuro-Fuzzy and Soft Computing, Prentice Hall, Upper Saddle River, NJ, 1997
- [4] J. Wesley Hines, Fuzzy and Neural Approaches in Engineering, John Wiley & Sons, 1997
- [5] John Hauser, Shankar Sastry, and Petar Kokotović, "Nonlinear Control Via Approximate Input-Output Linearization: The Ball and Beam Example", IEEE. Trans. Automatic Control, Vol.37, P392-398, March 1992