'95 대한전기학회 하계학술대회 논문집(1995, 7, 20~22)

퍼지-뉴럴 제어기를 이용한 유도전동기 속도제어

김 세찬*, 김 학성, 류 홍제, 원 충연
성균관 대학교 전기공학과

A Study on Induction Motor Speed Control Using Fuzzy-Neural Network

Kim Sei Chan*, Kim Hak Sung, Ryoo Hong Je, Won Chung Yuen
Dept. of Electrical Engineering, Sung kyun kwan University

ABSTRACT

The Fuzzy-Neural Controller is constructed to resolve some dificulties taking place in decision of membership functions,input and output gains and an inferenced method for desinging fuzzy logic controller. In addition Neural network emulator is used to emulate induction motor forward dynamics and to get error signal at fuzzy-neural controller output layer. Error signal is backpropagated through neural network emulator.

A back propagation algorithm is used to train fuzzy-neural controller and emulator. The experimental results show that this control system can provide good dynamical responses.

1.서 론

서이보 운전을 위한 제어 방법으로 산업현장에서 널리 쓰이고 있는 PI혹은 PID제어기는 제어상수값이 적절히 조절될 경우 좋은 운전 특성을 얻을 수 있으며 제어 알고리즘이 간단하여 손쉽게 구현할 수 있다. 그러나 운전점이나 파라메터가 변하는 경우 제어 상수 값을 변화 시켜야 하는 문제점이 발생한다[1]. 이러한 문제점을 해결하기 위하여 최근에 인간의 지능을 부여하는 전문가 지식을 기반으로 하는 퍼지 제어기, 신경망 제어기 또는 퍼지 로직과 신경망을 혼합한 형태인 퍼지-뉴럴 제어기를 자동제어 시스템에 적용하는 연구가 활발히 진행 되고 있다. 이러한 인공지능 제어기는 종래의 제어 알고리즘을 적용하여 얻은 결과 보다 우수한 결과를 나타냄이 증명 되고 있다[2]-[3].

퍼지 제어기는 플랜트의 수학적 모델링이 필요없고, 외란에 강인한 특성을 보여 준다. 그러나 제어규칙,소속함수 입출력 이득을 생성하기 위한 일반적 규칙이 존재하지 않고, 설계된 범위 밖의 입력 값에 대해서는 적절한 제어값을 생성하지 못하는 단점이 있다. 한편 신경망 제어기는 학습 기능,병렬 분산처리의 특성을 지니며 신경망 자체가 필터기능을 가지고 있어서 외란에 둔감한 특성을 지닌다[4]. 그러나 신경망 구조의 설계문제,학습방법에 있어서 이러움을 가지고 있다. 최근에는 이러한 퍼지제어기의 장점과 신경망의 장점을 결합한 형태의 퍼지-뉴럴제어기에 대한 연구가 활발히 진행 되고 있다[5]. 본 논문에서는 유도전동기 속도제어를 위한 퍼지-뉴럴제어기와, 제어기 출력단에서의 오차항을 구하기 위한 신경망 에뮬레이터를 구성하였다. 유도전동기의 실제 속도와 기준 속도와의 오차는 이 신경망 에뮬레이터를 통하여 역전파된다. 또한 빠른 명령주기를 가지는 DSP보드(TMS320C30)를 이용하여 매 샘플링 마다 오차를 역전파 시킴으로서 온라인 실시간 학습이 가능하게 하였다.

2.퍼지-뉴럴 제어기를 이용한 유도전동기 속도제어 시스템

본 논문에서는 퍼지 제어기의 제어규칙을 자동 생성하고 소속함수를 자동 조정하기 위하여 퍼지-뉴럴 제어기를 구성한다. 신경망의 형태로 구성된 퍼지 제어규칙들은 다음과 같이 크리스프한 결론부를 가지는 "IF~THEN~"의 형태를 가지고 있다.

$$\begin{aligned} & R_1: \text{ if } x_1 \text{ is } A_{11} \text{ and } x_2 \text{ is } A_{12} \text{ then } u \text{ is } C_1 \\ \text{or } & R_2: \text{ if } x_1 \text{ is } A_{21} \text{ and } x_2 \text{ is } A_{22} \text{ then } u \text{ is } C_2 \\ & \cdots\cdots\cdots\cdots \\ \text{or } & R_i: \text{ if } x_1 \text{ is } A_{i1} \text{ and } x_2 \text{ is } A_{i2} \text{ then } u \text{ is } C_i \\ & \cdots\cdots\cdots\cdots \\ \text{or } & R_n: \text{ if } x_1 \text{ is } A_{n1} \text{ and } x_2 \text{ is } A_{n2} \text{ then } u \text{ is } C_n \\ & \text{where } \quad x_1 \in U,\ x_2 \in V,\ u \in W \end{aligned} \tag{1}$$

여기서 x_1, x_2는 입력, u는 출력, A_{i1}, A_{i2},그리고 C_i는 그림 1과 같은 소속함수를 가지는 퍼지집합이다.

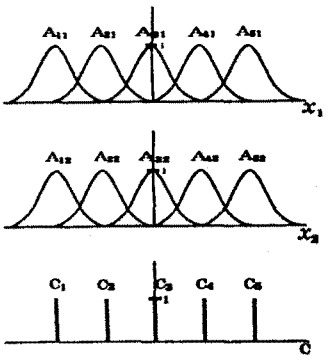

그림 1 소속함수의 모양
Fig. 1 Shapes of membership function

퍼지-뉴럴 제어기는 신경망의 형태로 구성된 퍼지 제어기이며 그림 2와 같이 구성된다.

그림 2에서와 같이 퍼지-뉴럴 제어기는 퍼지 제어기의 퍼지화에 해당하는 입력층 제어규칙을 구성하는 은닉층, 비퍼지화에 해당하는 출력층등의 3층 구조를 갖는다. 퍼지-뉴럴 제어기의 입력은

기준속도 ($r(k)$)와 실제속도와 ($y(k)$)의 오차

$$x_1(k) = se(k) = r(k) - y(k) \tag{2}$$

그리고 오차의 변화율

$$x_2(k) = sce(k) = se(k) - se(k-1) \tag{3}$$

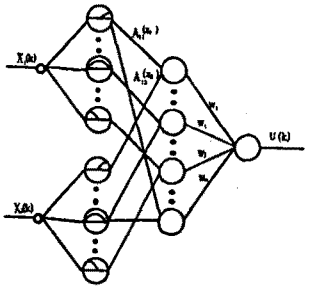

그림 2 퍼지-뉴럴 제어기

Fig. 2 Fuzzy-Neural controller

을 선택하였다. 그리고 각 층의 뉴런들은 상호 연결 되어 있으며 연결강도는 가중치 w_i로 표시되고 있다. 입력층에서는 각 입력 변수의 각 언어변수의 소속함수값에 해당하는 $A_{i1}(x_1)$,와 $A_{i2}(x_2)$를 출력한다.

여기서 사용된 각 입력에 대한 퍼지언어변수의 소속함수값은

$$A_{i1}(x_1)=\frac{1}{1+[\ (\frac{x_1-c_{i1}}{2a_{i1}})^2]^{\ b_{i1}}} \tag{4}$$

$$A_{i2}(x_2)=\frac{1}{1+[\ (\frac{x_2-c_{i2}}{2a_{i2}})^2]^{\ b_{i2}}} \tag{5}$$

에 의해 계산된다. 이 값들은 은닉층으로 전달되며 은닉층에서는 이 두 값을 곱한값

$$\mu_i=A_{i1}(x_1)A_{i2}(x_2) \tag{6}$$

를 출력하며 가중치 w_i와 곱해져 출력층으로 전달한다.

출력층은 비퍼지화 단계에 해당하므로 다음과 같은 식에 의해 무게중심법에 의해 비퍼지화를 실행한다.

$$y=\frac{\sum_i^n \mu_i w_i}{\sum \mu_i} \tag{7}$$

신경망 에뮬레이터는 그림 3과 같이 3개 입력층 뉴런,5개의 은익층 뉴런, 한 개의 출력층 뉴런을 가지는 3층의 다층 퍼셉트론으로 구성된다. 신경망 에뮬레이터의 입력으로는 퍼지-뉴럴제어기의 출력값 $u(k)$,그리고 퍼지-뉴럴 제어기의 입력으로 사용된 기준속도와 실제속도의 오차 $se(k)$,그리고 오차의 변화율 $sce(k)$을 선택하였다. 그림 4는 구성된 유도전동기 속도 제어 시스템을 보여준다.

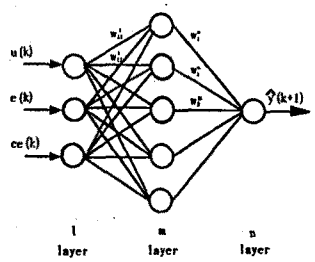

그림 3 신경망 에뮬레이터

Fig.3 Neural netork emulator

퍼지-뉴럴 제어기와 신경망 에뮬레이터의 학습과정은 다음과 같다.

먼저 신경망 에뮬레이터를 학습시키기 위하여 에뮬레이터의 출력과 유도전동기의 실제 출력과의 오차 $\hat{e}(k+1)$을 구하면

$$\hat{e}(k+1)=\frac{1}{2}(y(k+1)-\hat{y}(k+1))^2 \tag{8}$$

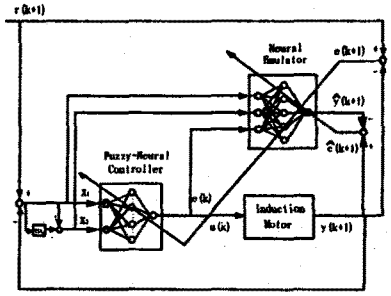

그림 4 유도전동기 속도제어 시스템

Fig.4 Speed control system of induction motor

; 여기서 $y(k+1)$은 유도전동기 실제 속도,

$\hat{y}(k+1)$는 신경회로망 에뮬레이터 출력

학습은 오차식 (8)을 최소화하도록 다음과 같이 일반화된 델타규칙에 의해 이루어진다.

$$\Delta W_i^m(k+1) \propto -\frac{\partial \hat{e}(k+1)}{\partial W_i^m} \tag{9}$$

신경망 에뮬레이터가 학습된 후 제어기 출력단에서의 오차항을 구하기 위하여 플랜트 출력단에서의 오차항 $e(k+1)$를 구하면

$$e(k+1)=\frac{1}{2}(r(k+1)-y(k+1))^2 \tag{10}$$

;여기서 $y(k+1)$는 유도전동기 실제 출력, $r(k+1)$ 는 속도 기준치.

이 오차항은 오차 역전파 알고리즘에 의해 신경회로망 에뮬레이터를 통하여 역전파 된다. 은닉층에서의 오차항은

$$\delta_i^m=f(net_i^m)\delta^m W_i^m=(1-(OUT_i^m)^2)\delta^m W_i^m \tag{11}$$

$$\delta^m = f(net^m)(r(k+1)-y(k+1)) \tag{12}$$

$$= (1-(y(k+1))^2)(r(k+1)-y(k+1))$$

이 되고 신경회로망 에뮬레이터 입력층에서 신경회로망 제어기의 출력 $u(k)$와 연결된 노드에서의 오차항은 은닉층의 각 노드에서 역전파 되어온 오차항의 합이 된다.

$$\delta_i^1 = \sum_j \delta_j^m \tag{13}$$

여기서 구해진 오차항 δ_j^1 은 다시 퍼지-뉴럴 제어기를 학습시키기 위한 오차항으로 사용된다. 한편 퍼지-뉴럴 제어기에서 학습될 파라메터는 식 (4), 의 a_{i1}, b_{i1}, c_{i1} 과 식 (5)의 a_{i2}, b_{i2}, c_{i2} 그리고 출력층에서의 가중치 ω_i가 된다. 각 파라메터는 오차 역전파 알고리즘에 의해 최적의 제어입력값을 출력하도록 실시간 학습된다. 각 층에서의 학습과정은 다음과 같다.

먼저 출력층에서의 가중치 변화량은

$$\Delta \omega_i \propto -\frac{\partial e(k)}{\partial \omega_i} \tag{14}$$

이 된다.

체인 룰(chain rule)에 의해 식(14)는

$$\Delta \omega_i = -\frac{\partial e(k)}{\partial \omega_i(k)} = -\frac{\partial e(k)}{\partial u(k)}\frac{\partial u(k)}{\partial \omega_i(k)} \tag{15}$$

로 표현될 수 있고 여기서

$$\frac{\partial u(k)}{\partial \omega_i(k)}=\frac{\partial}{\partial \omega_i(k)}(\frac{\sum \mu_i(k)\omega_i(k)}{\sum \mu_i(k)}) = \frac{\mu_i(k)}{\sum \mu_i(k)} \tag{16}$$

이 된다.

그러나 $\frac{\partial e(k)}{\partial u(k)}$ 는 $e(k)$ 를 구할 수 없으므로 계산이 불가능하다. 따라서 $\frac{\partial e(k)}{\partial u(k)}$ 를 계산하는 대신 이미 알고 있는 에뮬레이터를 통하여 역전파된 오차

값을 이용하면

$$\frac{\partial e(k)}{\partial u(k)} = \delta^{k+1} \tag{17}$$

$$\delta^{k+1} = \delta_j^j = \sum_j \delta_j^j \tag{18}$$

이 되며 이것을 사용하면 제어기 출력층에서의 가중치 변화량은

$$\Delta \omega_i(k+1) = \eta \delta^{k+1} \left(\frac{\mu_i(k)}{\sum \mu_i(k)}\right) \tag{19}$$

이 된다. 그러므로 출력층의 가중치 $\omega_i(k+1)$는

$$\omega_i(k+1) = \omega_i(k) + \Delta \omega_i(k+1) \tag{20}$$

에 의해 조정된다. 입력층의 소속함수 파라메터의 변화량 Δa_{ji}은

$$\begin{aligned} \Delta a_{ji} &= \frac{\partial e(k)}{\partial a_{ji}(k)} \\ &= \frac{\partial e(k)}{\partial u(k)} \frac{\partial u(k)}{\partial \mu_i(k)} \frac{\partial \mu_i(k)}{\partial A_{ji}(k)} \frac{\partial A_{ji}(k)}{\partial a_{ji}(k)} \\ &= -\eta \delta^{k+1} \frac{1}{\sum \mu_i(k)} * \\ &\quad (w_i(k) - u(k)) A_{2i}(x_2(k)) \frac{2}{a_{ji}(k)} [\, A_{ji}(k) - A_{ji}^2(k)] \end{aligned} \tag{21, 22}$$

가 된다. 그리고 파라메터 변화량 Δc_{ji}은

$$\begin{aligned} \Delta c_{ji} &= \frac{\partial e(k)}{\partial c_{ji}(k)} \\ &= \frac{\partial e(k)}{\partial u(k)} \frac{\partial u(k)}{\partial \mu_i(k)} \frac{\partial \mu_i(k)}{\partial A_{ji}(k)} \frac{\partial A_{ji}(k)}{\partial c_{ji}(k)} \\ &= -\eta \delta^{k+1} \frac{1}{\sum \mu_i(k)} * \\ &\quad (w_i(k) - u(k)) A_{2i}(x_2(k)) \frac{2}{a_{ji}(k)} [\, A_{ji}(k) - A_{ji}^2(k)] \end{aligned} \tag{23, 24}$$

유사하게

$$\begin{aligned} \Delta a_{2i} &= -\eta \delta^{k+1} \frac{1}{\sum \mu_i(k)} \\ &\quad (w_i(k) - u(k)) A_{2i}(x_2(k)) \frac{2}{a_{ji}(k)} [\, A_{ji}(k) - A_{ji}^2(k)] \end{aligned} \tag{25}$$

$$\begin{aligned} \Delta c_{2i} &= -\eta \delta^{k+1} \frac{1}{\sum \mu_i(k)} \\ &\quad (w_i(k) - u(k)) A_{2i}(x_2(k)) \frac{2}{a_{ji}(k)} [\, A_{ji}(k) - A_{ji}^2(k)] \end{aligned} \tag{26}$$

또한 구하여 진다.

소속함수의 파라메터 a_{ji}, b_{ji}, c_{ji} 과 a_{2i}, b_{2i}, c_{2i} 그리고 출력층에서의 가중치 w_i 의 학습률 (η)은 각각 0.1, 0.1, 0.9로 하였다.

3.실험결과 및 검토

그림 5는 유도전동기 속도제어 시스템의 하드웨어 구성도를 나타낸다. 유도전동기 속도제어 시스템의 하드웨어는 고성능 DSP TMS320C30보드와 IGBT 전류제어형 인버터로 구성된다[10]. 샘플링 주기는 2[ms]로 하였고, 그림 6과 같이 구성된 장치를 이용하여 M×L×sin(θ)[N·m]로 정현적으로 변화하는 부하에 대하여 부하실험을 행하였다. 그리고 표1.은 실험에서 사용된 유도전동기 파라메터이다.

표1. 실험에 사용된 전동기 상수 및 정격

전압 : 220[V]	출력 : 2.2[Kw]	극수 : 4[극]
회전수 : 1735[rpm]	주파수 : 60[Hz]	
R_s : 0.687[Ω]	L_s : 0.08397[H]	
R_r : 0.842[Ω]	L_r : 0.08525[H]	
M : 0.08136[H]	J : 0.03$Kg \cdot m^2$	
B : 0.01[$N \cdot m$/sec]		

그림 7 (a),(b)는 무부하에서 -500rpm ~ 500rpm으로 변화하는 기준속도를

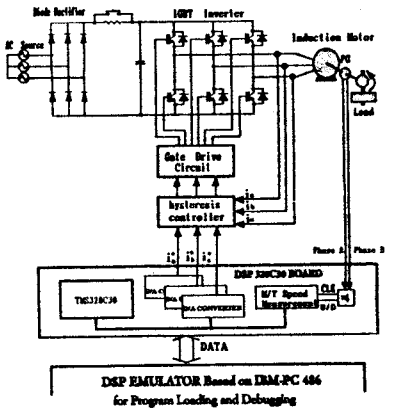

그림 5 하드웨어 구성도

Fig. 5 block diagram of hardware

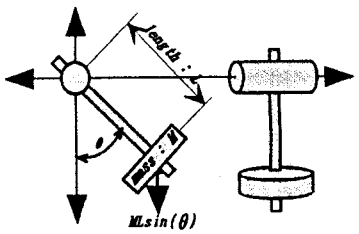

그림 6 전동기의 부하

Fig. 6 Load of motor

5회,10회 반복 인가하여 학습하였을 경우 유도전동기 실제속도와 기준속도를 보여준다. 5회 학습시에도 이미 기준속도를 잘 추종함을 알수 있고 학습이 진행될수록 오버슈터가 줄어들고 있음을 알 수 있다. 그리고 그림 8(a),(b)는 5회,10회 학습시의 유도전동기 실제 속도와 토크분 전류인 제어입력(u(k))을 보여준다. 그림 9(a),(b)는 $\sin(\theta)$[N.m]로 변화하는 부하를 인가하고 -500rpm - 500rpm으로 변화하는 기준속도를 5회, 10회 반복하여 학습하였을 경우 유도전동기 실제속도와 기준속도를 보여준다. 여기서 5회의 학습에도 기준 속도를 잘 추종함을 알수 있다. 그림 10(a),(b)는 유도전동기 실제 속도와 제어입력(u(k))을 보여준다.

4.결론

본 논문에서는 퍼지-뉴럴 제어기와 신경망 에뮬레이터를 이용하여 속도제어 시스템을 구성하여 다음과 같은 결론을 얻었다.

1) 신경망 에뮬레이터를 도입하여 제어기 출력단에서의 오차항을 계산하였다.
2) DSP보드를 이용하여 온-라인,실시간 학습 ,제어가 가능하게 함으로서 부하변동이나 외란에 실시간으로 적응할 수 있는 지능형 적응제어기를 구현하였다.
3) 실험결과 퍼지-뉴럴 제어기는 5회정도의 학습만으로도 기준 속도를 잘 추종함을 확인 하였고 정현적으로 변하는 부하변동에 대하여 강인성을 가짐을 확인하였다.

참 고 문 헌

1. K.Kenzo,O.Tsutomu,and S.Taskashi, "Application Trands in AC Motor Drives", IEEE IECON'92, pp.31-36, 1992.
2. 김세찬, "퍼지 제어기를 이용한 유도전동기 위치제어에 관한 연구", 성균관대학교 석사학위논문, 1992.
3. 김세찬, 김학성, 류홍제, 원충연, "신경회로망을 이용한 유도전동기 속도제어

에 관한 연구", 대한 전기학회 전력전자 연구회 춘계학술대회, pp.11-15, 1995. 5.

4. D.E.Rumelhart, J.L.McClelland, and The PDP Research Group, Parallel Distributed Processing. Vol.1-2, MIT Press, 1986.

5. 양승호 , "유도전동기 구동 시스템을 위한 뉴로-퍼지 제어기의 설계", 성균관 대학교 박사학위 논문, 1994.

6. Texas Instrument, TMS320C30x user's guide, 1990.

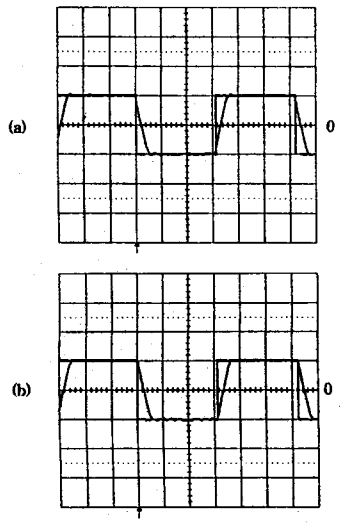

그림 7 유도전동기 실제속도와 기준속도
(a) 5회 학습시 (b) 10회 학습시
(x축 : 2 sec/div , Y축 : 500 rpm/div)
Fig. 7 Speed of induction motor and reference speed
(a) after 5 training (b) after 10 training
(x축 : 2 sec/div , Y축 : 500 rpm/div)

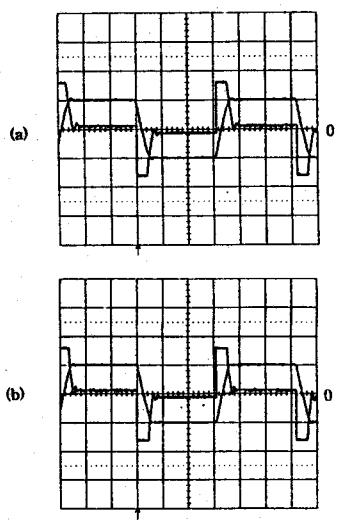

그림 8 유도전동기 속도와 제어입력
(a) 5회 학습시 (b) 10회 학습시
(x축 : 2 sec/div , Y축 : 500 rpm/div, 5 Λ/div)
Fig. 8 Speed of induction motor and control input
(a) after 5 training (b) after 10 training
(x축 : 2 sec/div , Y축 : 500 rpm/div, 5 Λ/div)

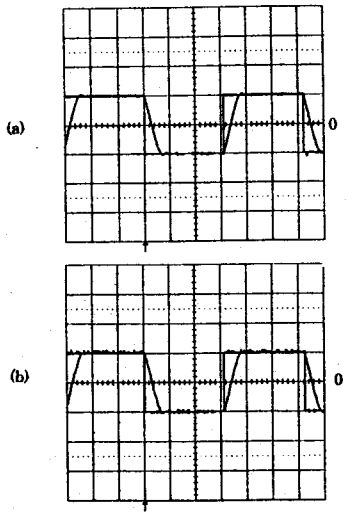

그림 9 부하인가시 유도전동기 속도와 기준출력
(a) 5회 학습시 (b) 10회 학습시
(x축 : 2 sec/div , Y축 : 500 rpm/div)
Fig. 9 Speed of induction motor and reference speed under load
(a) after 5 training (b) after 10 training
(x축 : 2 sec/div , Y축 : 500 rpm/div)

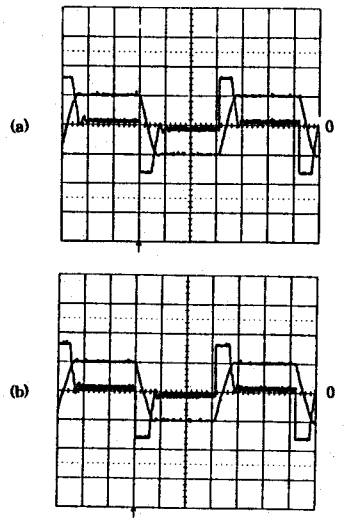

그림 10 부하인가시 유도전동기 실제출력과 제어입력
(a) 5회 학습시 (b) 10회 학습시
(x축 : 2 sec/div , Y축 : 500 rpm/div ,5 Λ/div)
Fig. 10 Speed of induction motor and control input under load
(a) after 5 training (b) after 10 training
(x축 : 2 sec/div , Y축 : 500 rpm/div ,5 Λ/div)