

## 신경회로망 알고리즘을 이용한 유도전동기 속도제어에 관한 연구

이 훈 구\*, 오 봉 환\*, 이 승 환\*\*, 전 기 영\*\*\*  
 \*용인송담대, \*명지전문대, \*\*대덕대학, \*\*\*명지대

### Study on Induction Motor Speed Control using Neural Network algorithm

H.G. Lee\*, B.H. Oh\*, S.H. Lee\*\*, K.Y. Jeon\*\*\*

\*Yong-in Songdam College, \*Myongji College, \*\*Taedok College, \*\*\*Myongji Univ.

**Abstract** - This paper presents a speed control system of induction motor using neural network. The speed control of induction motor was designed to NNC(Neural Network Controller) and NNE(Neural Network Estimator) used backpropagation, the NNE was constituted to be get an error value of output of an induction motor and conspire an input/output. NNC is controled to be made the error of reference speed and actual speed decrease, and in order to determine the weighting of NNC can be back propagated through the NNE, and it is adapted to the outside circumstances and system characters with learning ability.

출력인  $u(n)$ 은 지령 토오크분 전류  $i_{qs}$ 가 된다. 신경회로망 제어기의 입력으로는 지령속도  $\omega_r^*$ 과 지령속도와 실제속도의 오차 $e(n)$  그리고 오차변화를  $\Delta e(n)$ 로 선택하면 일반적인 오류 역전파 알고리즘을 이용할 수가 있다.

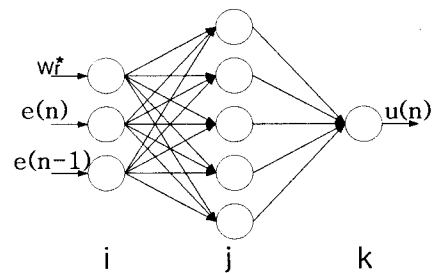


그림 1. 신경회로망 제어기

## 1. 서 론

유도전동기의 속도제어 방법으로는 PID제어기가 많이 사용되고 있으며 제어이득이 적절히 조정될 경우 좋은 운전 특성을 얻을 수 있다. 그러나 제어성능이 시스템의 파라미터 변동에 민감하여 최적의 제어이득을 구하기가 어렵고, 부하의란에 대해 빠른 과도응답을 갖지 못하는 단점이 있다. 따라서 이러한 문제점을 해결하기 위하여 학습능력을 가지고 외부환경이나 시스템특성에 적응할 수 있는 신경회로망 연구가 진행되고 있다. 신경회로망을 응용한 제어기인 경우 학습알고리즘으로 감독학습법을 대부분 사용하게 된다. 그러나 감독입력에 대한 감독출력값을 구하기가 어렵다. 따라서 감독입력출력을 구하는 문제가 신경회로망을 제어기로 사용할 경우에 매우 중요하다.<sup>[1]</sup>

그림 2는 신경회로망 속도추정기로 3개의 입력층 뉴런, 5개의 은닉층 뉴런, 한 개의 출력층 뉴런으로 구성한다. 입력은 신경회로망 제어기의 출력값  $u(n)$ , 그리고 지령속도와 실제속도의 오차분  $e(n)$ , 오차의 변화율  $\Delta e(n)$ 을 선택하여 가장 우수한 성능을 얻도록 하였다.

본 논문에서는 유도전동기의 속도제어를 오류역전파알고리즘을 이용한 NNC(Neural Network Controller)와 NNE(Neural Network Estimator)로 설계하였고, NNE는 유도전동기의 속도를 구한 후 실제속도와 비교해서 오차항을 구하여 감독 입력출력값을 생성할 수 있도록 구성하였다. 또한 지령속도와 실제속도의 오차가 최소가 되도록 NNC를 제어하고, NNC의 가중치는 오류 역전파 알고리즘을 이용하여 NNE의 오차항을 역전파하여 시스템의 특성 변화시 적응하도록 하였다.<sup>[2]</sup>

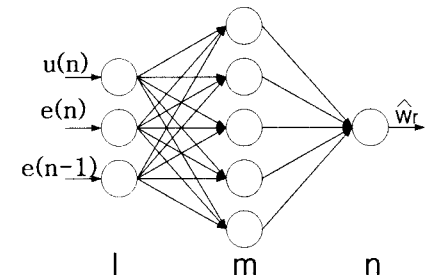


그림 2. 신경회로망 속도추정기

## 2. 신경회로망 구성

### 2.1 신경회로망의 시스템 구성

신경회로망 시스템은 학습시키고자 하는 입력성분과 출력성분 그리고 오차항의 가중치를 조절하기 위한 입력층, 은닉층, 출력층으로 구성된다. 그리고 신경회로망을 이용한 제어기와 속도추정기로 구성하고, 플랜트는 벡터 제어 인버터와 유도전동기로 구성하였다. 그림1은 신경회로망 제어기로 3개의 입력층 뉴런, 5개의 은닉층 뉴런, 1개의 출력층 뉴런인 다층 퍼셉트론으로 구성하고,

### 2.2 속도 제어시스템 구성

유도전동기의 속도제어에서 실제속도가 지령속도를 잘 추종할 수 있도록 신경회로망 제어기는 학습을 통하여 속도오차  $e(n) = (\omega_r^* - \omega_r)$ 가 최소가 되도록 제어값  $u(n)$ 을 발생시키는 것이 목적이다. 그리고 NNC를 학습시키기 위해서는 출력단의  $u(n)$ 을 구하여 오차 평가 함수가 최소가 되도록 가중치를 조절 해야 한다. 그러나 NNC를 구성하였을 경우 출력단에서의 오차항  $eu(n)$ 은 감독출력값  $u_r(n)$ 가 없으므로 구할 수 없다. 그러므로  $u_r(n)$ 를 구하지 않고 NNE를 통하여 유도전동기의 출력단의 오차항을 역전파함으로써, NNC 출력단에서의 오차항을 구한다. NNE의 역할은 유도전동기를 근사화

하도록 학습하고, 유도전동기 출력단에서 오차항을 역전파시켜 NNC출력단에서의 오차항을 계산할 수 있도록 하였다. 또한 신경망제어기와 속도추정기의 입력층의 입력범위가  $\pm 1$ 의 값을 가진 활성화 함수인 양방향 시그모이드함수인  $\tanh(\cdot)$ 를 사용하였다. 그림 3은 신경회로망을 이용한 유도전동기의 속도 제어 블록도를 나타낸다.

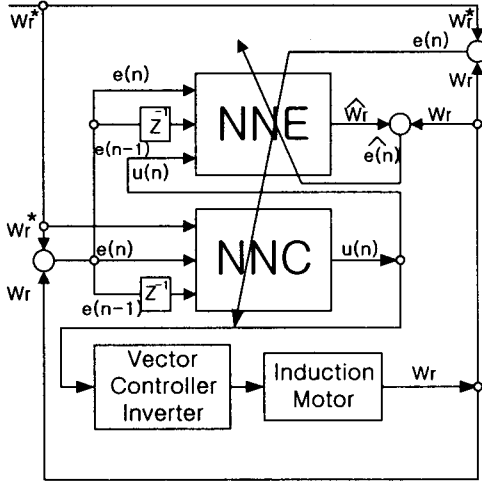


그림 3. 신경회로망을 이용한 속도 제어블록도

### 2.3 신경회로망의 학습 알고리즘

NNE의 역할은 유도전동기를 근사화 하도록 학습하고, 유도전동기의 출력단에서 오차항을 역전파시켜 NNC출력단에서의 오차항을 계산할 수 있도록 하기 위하여 실제속도와 추정속도의 오차함수를 구하면 식(1)과 같다.

$$\hat{e}(n) = \frac{1}{2} (\omega_r^* - \hat{\omega}_r)^2 \quad (1)$$

오차가 최소가 되도록 하기 위하여 식 (1)을 일반화한 델타규칙에 의해 식 (2)와 같이 나타낸다.

$$\Delta W_j^m(n) = - \frac{\partial \hat{e}(n)}{\partial W_j^m} \quad (2)$$

식 (2)을 체인규칙에 의해 우변을 식 (3)과 같이 나타낼 수 있다.

$$- \frac{\partial \hat{e}(n)}{\partial W_j^m} = \frac{\partial \hat{e}(n)}{\partial net^n} \frac{\partial net^n}{\partial W_j^m} \quad (3)$$

식(3)을 나누어 정리하면 식 (4)와 (5)로 나타낸다.

$$\frac{\partial net^n}{\partial W_j^m} = \frac{\partial}{\partial W_j^m} \sum_i OUT_i^m W_i^m = OUT_j^m \quad (4)$$

$$\delta^n = - \frac{\partial \hat{e}(n)}{\partial net^n} \quad (5)$$

식 (4)와 (5)에 의해 식 (3)을 다시 정리하면 식 (6)과 같이 나타낸다.

$$- \frac{\partial \hat{e}(n)}{\partial W_j^m} = \delta^n OUT_j^m \quad (6)$$

식 (6)에서  $\delta^n$ 은 출력층 뉴런의 오차항이며 체인규칙에 의해 식 (7)과 같이 나타낸다.

$$\delta^n = - \frac{\partial \hat{e}(n)}{\partial net^n} = - \frac{\partial \hat{e}(n)}{\partial OUT^n} \frac{\partial OUT^n}{\partial net^n} \quad (7)$$

식 (7)을 정리하면 식 (8), (9)이 된다.

$$\frac{\partial OUT^n}{\partial net^n} = f'(net^n) \quad (8)$$

$$OUT^n = f(net^n) = \tanh(net^n) \quad (9)$$

그리고, 식 (7)의 우변의 경우, 식 (10)으로 나타낼 수 있다.

$$\frac{\partial \hat{e}(n)}{\partial OUT^n} = -(\omega_r - \hat{\omega}_r) \quad (10)$$

그러므로, 출력층 뉴런의 오차항은 식 (11)과 같이 나타내고, 은닉층 뉴런의 오차항은 감독 출력값을 알 수 없으므로 출력층의 오차항을 이용하여 구하면 식 (12)와 같이 나타낸다.

$$\begin{aligned} \delta^n &= f'(net^n)(\omega_r - \hat{\omega}_r) \\ &= (1 - \hat{\omega}_r^2)(\omega_r - \hat{\omega}_r) \end{aligned} \quad (11)$$

$$\begin{aligned} \delta_j^m &= f'(net_j^m) \delta^n W_j^m \\ &= [1 - (OUT_j^m)^2] \delta^n W_j^m \end{aligned} \quad (12)$$

오차  $\hat{e}(n)$ 에 대한 가중치의 조절은 보다 빠른 학습을 위해서 모멘텀항이 추가되어 식 (13)과 같이 나타낸다.

$$\Delta W_j^m(n+1) = \eta \delta^n OUT_j^m + \alpha \Delta W_j^m(n) \quad (13)$$

단,  $\alpha$  ( $0 < \alpha < 1$ )는 관성항이다.

NNE가 학습된 후 NNC의 오차항을 구하면 식 (14)와 같이 나타낸다.

$$e(n) = \frac{1}{2} (\omega_r^* - \omega_r)^2 \quad (14)$$

식 (14)에서 NNE의 오차항이 오류 역전과 알고리즘에 의해 역전과 된다. 그리고 식 (15), (16)은 NNC의 은닉층과 출력층의 오차항을 나타낸다.

$$\begin{aligned} \delta_j^m &= f'(net_j^m) \delta^n W_j^m \\ &= [1 - (OUT_j^m)^2] \delta^n W_j^m \end{aligned} \quad (15)$$

$$\begin{aligned} \delta^n &= f'(net^n)(\omega_r^* - \omega_r) \\ &= (1 - \omega_r^2)(\omega_r^* - \omega_r) \end{aligned} \quad (16)$$

NNC의 출력  $u(n)$ 과 연결된 노드의 오차항은, NNE 입력층과 연결되어 있으므로, 오차항은 NNE 은닉층의 각 노드에서 역전과 되어온 오차의 합이 된다. 그러므로 NNE에서 역전과 되어온 오차항은 NNC의 오차항  $\delta_j^i$ 이 되고,  $\delta_j^i$ 은 NNC를 학습시키기 위한 오차항이 된다. NNC 출력층의 가중치와  $\delta_j^i$ 는 식 (17), (18)과 같이 나타낸다.

$$\delta_j^i = \sum_j \delta_j^m \quad (17)$$

$$\Delta W_j^i = - \frac{\partial ue(n)}{\partial W_j^i} \quad (18)$$

그리고, 식 (3)~(10)과 같은 방법으로 식 (18)을 정리하면, NNC출력단의 가중치 변화량은 식 (19)와 같이 나타낸다.<sup>[3]</sup>

$$\begin{aligned} \Delta W_j^i &= - \frac{\partial ue(n)}{\partial W_j^i} = - \frac{\partial ue(n)}{\partial net^n} \frac{\partial net^n}{\partial W_j^i} \\ &= - \frac{\partial ue(n)}{\partial u(n)} \frac{\partial u(n)}{\partial net^n} \frac{\partial net^n}{\partial W_j^i} \\ &= \delta^{n+1} f'(net^n) OUT_j^i \end{aligned} \quad (19)$$

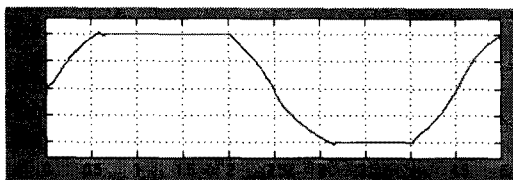
### 3. 시뮬레이션 및 실험결과

유도전동기의 속도제어에서 지령속도를 추정하기 위해서는 토크분 전류  $i_{qs}$ 를 제어하여야 한다. 본 논문에서 토크분 전류  $i_{qs}$ 가 신경망제어기의 출력인  $u(n)$ 의 값으로 공간벡터제어기로 들어간다. 공간벡터제어기는 토크분 전류와 자속전류로부터 슬립각속도를 계산하고 회전자 속도를 더하여 단위 벡터를 얻는 간접 벡터제어 방식을 사용하였다. 신경회로망 제어기는 주어진 지령속도를 잘 추종할 수 있도록 학습에 의하여 가중치를 조절하게 된다. 신경회로망 제어기의 오차항은 신경회로망 추정기에서 역전파된 오차항을 이용하여 학습하게 된다. 신경회로망 제어기와 신경회로망 추정기의 학습률은 각각 0.2와 0.5로 하고 활성화 함수는 입력범위가  $\pm 1$ 의 값을 가진 활성화 함수인 양방향 시그모이드함수인  $\tanh(\cdot)$ 를 사용하였다. 신경회로망의 구성은 3개의 입력층과 5개의 은닉층, 1개의 출력층을 갖는 다중 퍼셉트론으로 구성하였다. 표 1은 시뮬레이션에 사용한 유도전동기의 정격과 파라미터를 나타낸다.

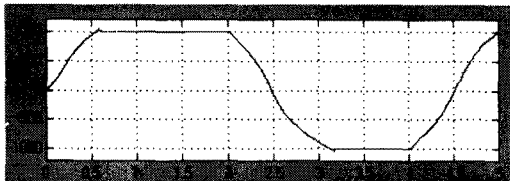
표1. 유도전동기의 정격

출 력	5[Hp]
정 격 전 압	3상 220[V], 60[Hz]
정 격 전 류	15[A]
정 격 회 전 수	1750[RPM]
$R_s$	0.407[Ω]
$R_r$	0.445[Ω]
$L_s$	73[mH]
$L_r$	70[mH]
$L_m$	67[mH]
$J$	0.079[ $\text{kg} \cdot \text{m}^2$ ]
$a$	0.0052[ $\text{Kg} \cdot \text{m}^2/\text{sec}$ ]

그림 4의 (a)와 (b)는 지령속도를 0[rpm]에서 800[rpm]으로, 다시, -800[rpm]에서 800[rpm]으로 스텝기동시의 추정 속도파형과 실제 속도파형을 나타낸다. (a)와 (b)에서처럼 추정속도파형이 실제 속도파형과 일치함을 확인 할 수 있다.



(a) 추정 속도파형



(b) 실제 속도파형

그림 4. 속도 추정파형과 실제 속도파형

그림 5는 지령속도를 0[rpm]에서 250[rpm]으로, 다시, -250[rpm]에서 250[rpm]으로 변화시의 NNC의 출력  $u(n)$  과 속도 추정파형을 나타낸다. 그림과 같이 신경망 회로망 제어기 출력인  $u(n)$ 이 지령 토크분 전류로서 지령속도를 잘 추종하는 것을 알 수 있다.

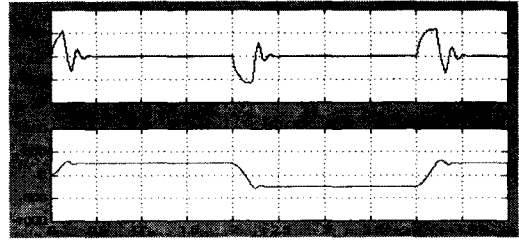


그림 5. NNC의 출력과 속도 추정파형

그림 6은 실제속도파형과 NNC에 의한 속도 추정파형을 나타낸다. 실제 실험파형과 추정파형이 거의 일치함을 확인 할 수 있다. 그림 7은 NNC에 의한 속도 추정파형과 전류파형을 나타낸다. NNC의 출력에 의한 추정속도파형이 실제속도를 양호하게 추정함을 확인할 수 있었다.

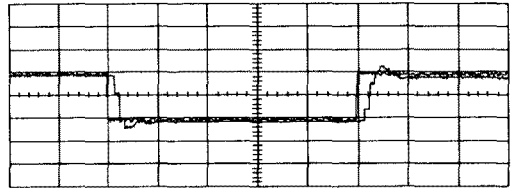


그림 7. 실제속도파형과 추정속도파형

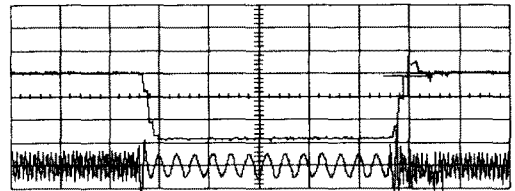


그림 8. NNC의 속도 추정파형과 전류파형

### 4. 결 론

본 논문은 신경회로망 제어기를 이용한 속도추정기 구성하여 다음과 같은 결론을 얻었다.

- [1] NNC 출력단 오차항을 NNE의 오차항을 역전파해서 감득 출력값  $u(n)$ 을 구하므로 신경회로망 구조를 간단하게 구성할 수 있었다.
- [2] 유도전동기는 상태변수와 파라미터가 서로 연관된 다변수 비선형 시스템으로서 속도제어가 어렵지만, 신경회로망의 입출력 데이터를 이용한 오류역전파 알고리즘 학습방법으로 속도를 제어하여, 속도추정이 양호함을 확인할 수 있었다.

### [참 고 문 헌]

- [1] 김세찬, 원충연, "신경회로망을 이용한 유도전동기의 속도 제어", KIEE, Vol.45, No.1, pp45~53, 1996
- [2] Toshio Fukuda, Talanora Shibata, " Theory and Applications of Neural Network for Industrial Control System" IEEE Trans. on Industrial Elec., Vol. 39, No. 6, Dec.,
- [3] J. M. Zurada, Induction to Artifical Neural System, West Publishing Company 1992