

## FNN과 NNC를 이용한 SynRM 드라이브의 고성능 속도제어

김순영\*, 고재섭\*, 강성준\*, 장미금\*\*, 문주희\* 이진국\*, 정동화\*  
 순천대학교\*, 전자부품연구원\*\*

### High Performance Speed Control of SynRM Drive using FNN and NNC

Soon-Young Kim\*, Jae-Sub Ko\*, Seong-Jun Kang\*, Mi-Geum Jang\*\*, Ju-Hui Mun\*, Jin-Kook Lee\*, Dong-Hwa Chung\*  
 Suncheon National University\*, Korea Electronics Technology Institute\*\*

**Abstract** - This paper is proposed design of high performance controller of SynRM drive using FNN and NNC. Also, This paper is proposed of designing fuzzy neural network controller(FNNC) which adopts the fuzzy logic to the artificial neural network(ANN). FNNC combines the capability of fuzzy reasoning in handling uncertain information and the capability of neural network in learning from processes. This controller is controlled speed using FNNC and model reference adaptive fuzzy control(MFC), and estimation of speed using ANN. The performance of proposed controller was demonstrated through response results. The results confirm that the proposed controller is high performance and robust under the variation of load torque and parameters.

#### 1. 서 론

산업 프로세서의 적용에는 부하 외란, 파라미터 변화 및 어떤 모델의 불확실성 등에 대하여 더욱 고성능과 강인성을 발휘할 수 있는 기법이 필요하다. 따라서 최근에는 적응 FNN(Fuzzy-Neural Network)과 ANN(Artificial Neural Network)이용[1], 하이브리드 인공지능제어[2], NN(Neural Network) PI 제어[3] 및 HB(Hybrid) PI 제어[4] 등의 연구가 활발하게 진행되었다. 이러한 연구들은 SynRM 드라이브에 적용하여 양호한 성능을 얻을 수 있었다. 그리고 이 연구들은 고속 및 저속의 속도변화, 부하 변동, 관성 변화 및 정·역회전 운전 등 다양한 동작에도 우수한 특성을 얻을 수 있었다. 그러나 FLC(Fuzzy Logic Controller)와 NNC(Neural Network Controller)는 직렬형태로 상호 의존관계로만 사용하였으며 병렬형태의 상호 독립적인 관계에서는 사용하지 않았다. 따라서, 본 논문에서는 FNNC(Fuzzy Neural Network Controller)를 이용하여 SynRM의 드라이브의 속도를 제어한다. 본 논문에서 제시한 FNNC의 제어특성을 분석하고 그 결과를 통해 타당성을 입증한다.

#### 2. FNNC 제어기

##### 2.1 NNC의 구조

SynRM의 방정식은 연속적인 시간영역에서 SISO(Single Input Single Output)로 표현할 수 있다.

$$L_{qs}J \frac{d^2 \omega_r(t)}{dt^2} + (RJ + L_{qs}B + K_2L_{qs}) \frac{d\omega_r(t)}{dt} + (RB + K_bK_T + K_2R)\omega_r(t) + K_1L_{qs} \frac{d\omega_r^2(t)}{dt} + K_1R\omega_r^2(t) + K_3R - K_TV_{qs}(t) = 0 \quad (1)$$

이산형태로  $q$ 축 전류를 나타내면 다음 식과 같다.

$$i_{qs}(k) = A_1 i_{qs}(k-1) + B_1 [\omega_r(k) - (\alpha + \frac{\epsilon C}{B}) \omega_r(k-1) - \beta \omega_r(k-2) - \gamma \omega_r^2(k) - \delta \omega_r^2(k-2) - \theta] / \epsilon \quad (2)$$

여기서,

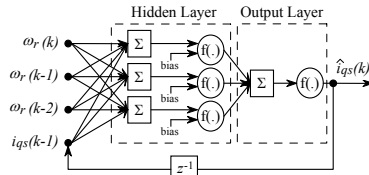
$$\begin{aligned} \alpha &= [2L_{qs}J + \Delta T(RJ + L_{qs}B + K_2L_{qs}) - \Delta T^2(RB + K_bK_T + K_2L_{qs})] / D \\ \beta &= -L_{qs}J / D & \gamma &= -[\Delta TK_1(L_{qs} + R\Delta T)] / D \\ \delta &= [\Delta TK_1L_{qs}] / D & \epsilon &= [K_T\Delta T^2] / D \\ \theta &= -[K_3R\Delta T^2] / D & D &= L_{qs}J + \Delta T(RJ + L_{qs}B + K_2L_{qs}) \end{aligned}$$

$$A_1 = 1 - \frac{R\Delta T}{L_{qs}}, B_1 = \frac{\Delta T}{L_{qs}}, C = -\frac{\Delta TK_b}{L_{qs}}$$

NNC 형태에서 동정모델은 다음 식으로 표현할 수 있다.

$$\hat{i}_{qs}(k) = NNC[i_{qs}(k-1), \omega_r(k), \omega_r(k-1), \omega_r(k-2)] \quad (3)$$

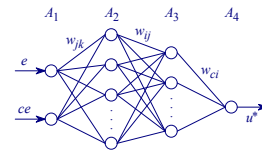
NNC 구조는 그림 1과 같다.



<그림 1> NNC 구조

##### 2.2 FNN 구조

그림 2는 FNN의 구성을 나타내며 여기서 두 개의 입력변수는 오차  $e$ 와 오차의 변화분  $ce$ 이고 출력변수는 제어변수  $u^*$ 이다.



<Fig. 2> The construction of FNN

##### 2.2.1 제어기의 조건부의 구현

그림 1에서  $A_1$ 층에서  $A_3$ 층 사이에 신경회로망은 퍼지물의 조건부의 구현을 나타낸다. 제어기의 조건부에서의 가중치 조절은 다음과 같다.

$$W_{ij}(t+1) = W_{ij}(t) + \Delta W_{ij} + \alpha[W_{ij}(t) - W_{ij}(t-1)] \quad (1)$$

$$W_{jk}(t+1) = W_{jk}(t) + \Delta W_{jk} + \alpha[W_{jk}(t) - W_{jk}(t-1)] \quad (2)$$

##### 2.2.2 제어기의 결론부의 구현

그림 1에서  $A_3$ 층과  $A_4$ 층 사이의 신경회로망은 퍼지물의 후반부의 구현을 나타낸다. 학습 동안 가중치  $W_{ci}$ 는 다음 오차 함수를 최소화하기 위하여 조절한다.

$$E^* = \frac{1}{2} \sum (U^* - U)^2 \quad (3)$$

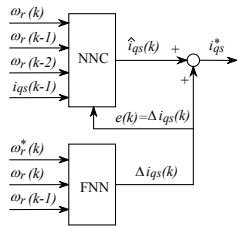
일반적인 델타룰을 사용하여 가중치  $W_{ci}$ 의 변화  $\Delta W_{ci}$ 는 오차함수인 식(3)을 최소화할 수 있고 퍼지물의 후반부를 재 정의하여 다음 식을 결정한다.

$$\Delta W_{ci} = -\eta \frac{\partial E^*}{\partial W_{ci}} = -\eta \delta_c O_i \quad (4)$$

$$W_{ci}(t+1) = W_{ci}(t) + \Delta W_{ci} + \alpha[W_{ci}(t) - W_{ci}(t-1)] \quad (5)$$

여기서  $\delta_c$ 는 퍼지와 신경회로망의 출력에서 오차신호이다.

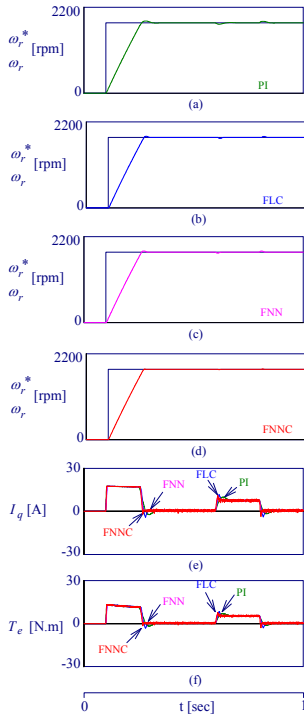
그림 3은 FNNC의 구조를 나타낸다.



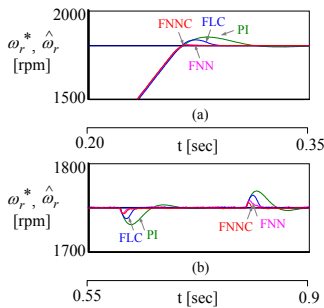
〈그림 3〉 FNNC의 구조

### 3. 시스템 성능결과

그림 4는 0.1[sec]에 1800[rpm]으로 운전중 0.6[sec]에서 0.8[sec]에 부하토크를  $5[N.m]$  인가하였을 때 응답특성을 나타낸다. 그림 4(a)는 PI 제어기, 그림 4(b)는 FLC, 그림 4(c)는 FNN 제어기, 그림 4(d)는 본 논문에서 제시한 FNNC 제어기의 지령속도와 실제속도를 나타낸다. 그림 4(e)는 q축 전류, 그림 4(f)는 발생토크를 나타낸다. 그림 5는 그림4의 과도상태 부분을 명확하게 해석하기 위하여 확대한 그림을 나타낸다. 그림 5(a)는 속도상승부분, 그림 5(b)는 부하변화 부분을 나타낸다. 본 논문에서 제시한 FNNC 제어기가 종래의 PI, FLC, FNN 제어기보다 오버슈트가 작고 안정화 시간이 빠르게 나타나고 있다.

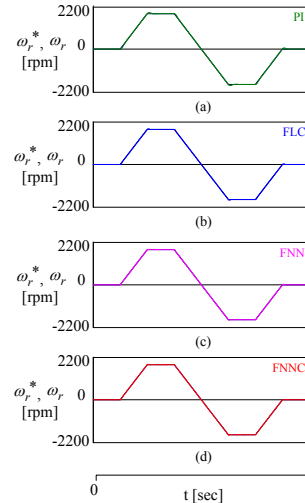


〈그림 4〉 지령속도 및 부하변화에 대한 응답특성

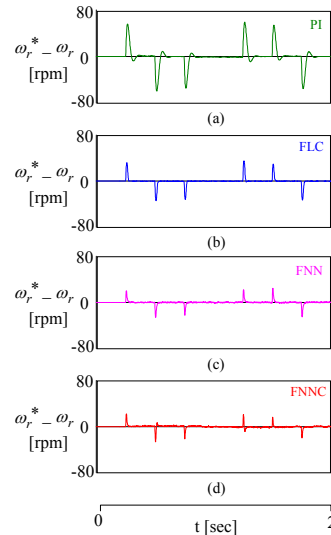


〈그림 5〉 지령속도 및 부하변화에 대한 응답특성(그림4확대)

그림 6은 4상한 운전에 대한 속도응답특성을 나타내고, 그림 7은 그림 6의 4상한 운전에 대한 속도오차를 나타낸다. 그림 7에서 본 논문에서 제시한 FNNC 제어기의 속도오차가 가장 작게 나타나고 있어 우수한 응답특성을 나타낸다.



〈그림 6〉 4상한 운전에 대한 응답특성(속도응답)



〈그림 7〉 4상한 운전에 대한 응답특성(속도오차)

### 4. 결 론

본 논문에서는 FNN과 NNC를 이용하여 SynRM 드라이브의 고성능 속도 제어기를 설계하였다. FNNC는 FNN과 NNC를 병렬구조로 연결하여 전동기가 정상상태일 경우 상호 독립적으로 제어가 이루어진다. FNNC 제어기는 종래의 PI, FLC, FNN 제어기와 속도, 부하 등의 변화에 대하여 응답특성을 비교, 분석하였으며 본 논문에서 제시한 FNNC 제어기가 오버슈트, 안정화시간, 속도오차 등의 부분에서 더욱 양호하게 나타났다. 따라서 본 논문의 타당성을 입증하였다.

### [참 고 문 헌]

- [1] H. G. Lee, S. M. Nam, J. S. Ko, J. S. Choi, J. C. Lee and D. H. Chung, "The speed control and estimation of IPMSM using adaptive FNN and ANN," ICCAS 2005, p. 134, 2005.
- [2] J. S. Ko, J. S. Choi, J. H. Lee and D. H. Chung, "Maximum torque control of IPMSM drive with hybrid artificial intelligent controller," Proceeding of ICMATE'06, Session B1, pp. 177-182, 2006.
- [3] J. S. Choi, J. S. Ko, J. H. Lee and D. H. Chung, "Speed control of IPMSM drive using neural network PI controller," ICEE '06, pp. 102, 2006.
- [4] J. S. Ko, J. S. Choi, K. T. Park, B. S. Park and D. H. Chung, "Development of HBPI Controller for High Performance Control of IPMSM Drive", pp. 368-372, ICPE'07, 2007.