

# 퍼지 보상기를 사용한 리커런트 시간지연 신경망 제어기 설계 Design of Recurrent Time Delayed Neural Network Controller Using Fuzzy Compensator

이상윤\*, 한성현\*, 신위재\*\*

\*경남대학교 기계자동화공학부, \*\*경남대학교 전기전자공학부

## ABSTRACT

In this paper, we proposed a recurrent time delayed neural network controller which compensate a output of neural network controller. Even if learn by neural network controller, it can occur an bad results from disturbance or load variations. So in order to adjust above case, we used the fuzzy compensator to get an expected results. And the weight of main neural network can be changed with the result of learning a inverse model neural network of plant, so a expected dynamic characteristics of plant can be got. As the results of simulation through the second order plant, we confirmed that the proposed recurrent time delayed neural network controller get a good response compare with a time delayed neural network controller.

## I. 서론

초기 신경망의 응용분야는 패턴 인식, 음성 인식, 화상 판단과 같은 신호처리 분야에 집중되었으나, 비선형 시스템을 모델링하고 근사화시킬 수 있는 능력으로 인해 현재 많은 제어시스템에 적용되고 있다.[1][2]

신경망은 비선형 시그모이드 활성화함수를 사용하여 전통적 제어법으로 접근하지 못하는 난해한 비선형 제어 문제를 해결하기 위한 비선형 사상능력을 제공해 준다.[2] 하지만, 학습시간이 많이 소요되고, 학습에 필요한 플랜트의 입·출력 데이터를 필요로 한다. 또한, 학습이 완료되어진 신경망을 사용하더라도 예상치 못한 외란으로 인해 플랜트의 출력이 좋지 못한 경우도 있다.

퍼지제어기는 제어대상에 대한 작업자의 경험이나 전문가의 언어적인 표현을 쉽게 제어기와 결합시킬 수 있는 장점을 지니고 있다. 또한 플랜트 파라미터 변동이나 외부 외란에 크게 영향을 받지 않는 강인한 제어기의 특성도 지니고 있다.[3]

본 논문에서는 신경망과 퍼지이론의 장점을 이용하여 학습된 신경망제어기의 플랜트 출력이 외란으로 인해

좋지 못한 결과를 가져올 때, 이를 퍼지제어기를 응용한 퍼지보상기를 사용하여 원하는 결과를 얻을 수 있도록 하는 제어기[4]에서 사용한 시간지연 신경망을 리커런트 시간 지연 신경망으로 대체하여 성능이 개선됨을 보였다.

## II. 신경망 제어기 설계

신경망 제어기는 주 제어기인 신경망제어기와 신경망 제어기의 출력을 보상시켜주는 퍼지보상기, 플랜트의 역모델을 학습하는 역모델 신경망(Inverse Model Neural Network: IMNN)으로 구성된다.(그림 1)

신경망제어기는 기준 입력  $r(n)$ 와 과거의 플랜트의 입력  $u_p(n-1)$ 을 이용하여 제어입력  $u_{nnc}(n)$ 을 생성한다.

퍼지보상기는 기준 입력  $r(n)$ 과 과거의 플랜트의 출력  $y(n-1)$ 사이의 오차  $e_{flc}(n)$ 와 오차 변화율  $\Delta e_{flc}(n)$ 을 사용하여 또 다른 제어입력  $u_{flc}(n)$ 을 계산한다.

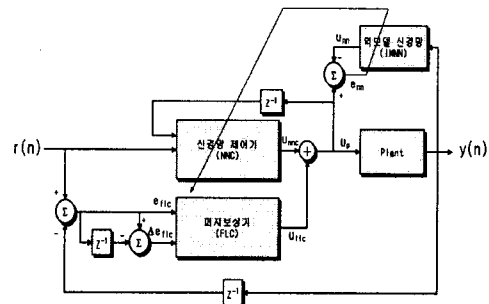


그림 1. 퍼지 보상기를 가지는 신경망 제어기 구조.

Fig. 1. The structure of neural network controller within fuzzy compensator.

$$e_{flc}(n) = r(n) - y(n-1) \quad (1)$$

$$\Delta e_{flc}(n) = e_{flc}(n) - e_{flc}(n-1) \quad (2)$$

신경망제어기의 출력  $u_{nnc}(n)$ 와 퍼지보상기의 출력

$u_{flc}(n)$ 의 합이 플랜트의 입력  $u_p(n)$ 가 된다.

$$u_p(n) = u_{nnc}(n) + u_{flc}(n) \quad (3)$$

플랜트의 출력  $y(n)$ 을 입력으로 한 역모델 신경망의 출력  $u_{nn}(n)$ 와 플랜트 입력  $u_p(n)$ 사이의 오차  $e_{nn}(n)$ 을 이용하여 역모델 신경망을 학습시킨 후, 역모델 신경망의 가중치를 신경망제어기의 가중치로 대체시킨다. 이렇게 함으로써, 시간이 경과함에 따라 신경망제어기의 성능을 향상시킬 수 있다.

### 2.1 신경망 구조와 학습법

신경망 제어기에는 시간지연 신경망(Time Delayed Neural Network: TDNN)[5][6]을 개선한 리커런트 시간지연 신경망(Recurrent Time Delayed Neural Network: RTDNN)을 사용하였으며, 역모델 신경망에는 NARX(nonlinear autoregressive with exogenous inputs)[6] 신경망을 개선한 리커런트 NARX(Recurrent NARX)를 사용하였다. 두 신경망 모두 시간지연과 리커런트 기능을 갖기 때문에, 신경망 구조를 차분방정식 또는 순환방정식으로 기술할 수 있다. 따라서, 시간에 따라 변하는 비선형 동적 시스템을 모델링하기에 유리하다. 그리고 플랜트 역모델을 온-라인으로 학습시킬 수 있기 때문에, 학습이 부족한 신경망제어기의 성능을 온-라인으로 보완시킬 수 있다.

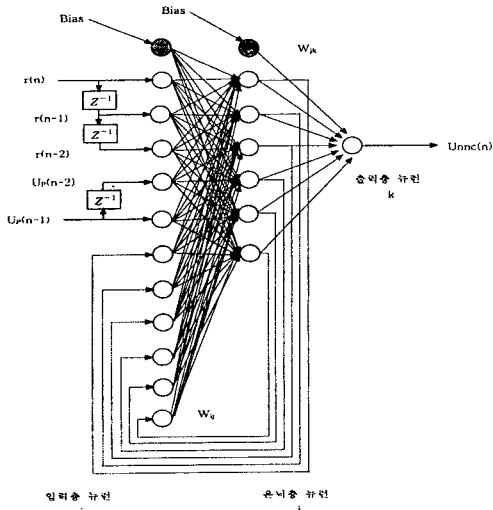


그림 2. 신경망제어기로 사용된 리커런트 시간지연신경망.

Fig. 1. The neural network controller consist of Recurrent TDNN.

학습 알고리즘은 일반적으로 신경망을 학습시키는데 사용하는 오류 역전파 학습 알고리즘을 사용하였다. 그리고 지역 극소점에 빠지는 것을 방지하고 빠른 학습을 위하여 모멘텀법을 사용하였으며 가중치 조절식은 다음과 같다.

$$\Delta W_{ij}(n+1) = 2\eta e_{nn}(n)y(n)(1-y(n))h(n) \frac{(1-h(n))W_{jk}(n)x(n)}{+ \alpha \Delta W_{ij}(n)} \quad (4)$$

$$\Delta W_{jk}(n+1) = 2\eta e_{nn}(n)y(n)(1-y(n))h(n) \frac{+ \alpha \Delta W_{jk}(n)}{+ \alpha \Delta W_{jk}(n)} \quad (5)$$

$$W_{ij}(n+1) = W_{ij}(n) + \Delta W_{ij}(n+1) \quad (6)$$

$$W_{jk}(n+1) = W_{jk}(n) + \Delta W_{jk}(n+1) \quad (7)$$

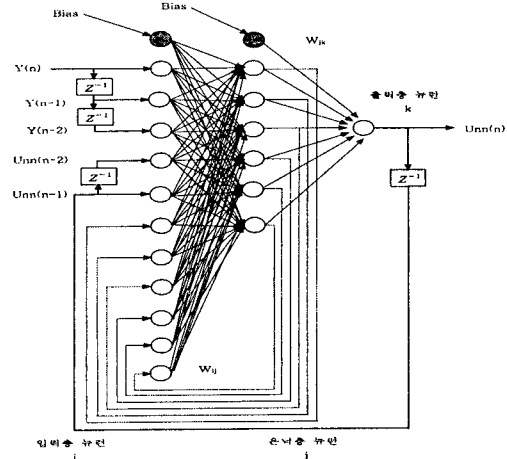


그림 3. 역모델 신경망에 사용된 Recurrent NARX 신경망.

Fig. 3. The Inverse model of neural network used to the Recurrent NARX neural network.

본 논문에서는 학습률  $\eta$ , 모멘텀 상수  $\alpha$ 를 각각 0.4와 0.7로 설정하였다.  $x(n)$ ,  $h(n)$ ,  $y(n)$ 는 각각 입력층 뉴런값, 은닉층 뉴런값, 출력층 뉴런값을 나타내며  $W_{ij}(n)$ 는 입력층과 은닉층사이의 가중치,  $W_{jk}(n)$ 는 은닉층과 출력층사이의 가중치를 나타낸다.

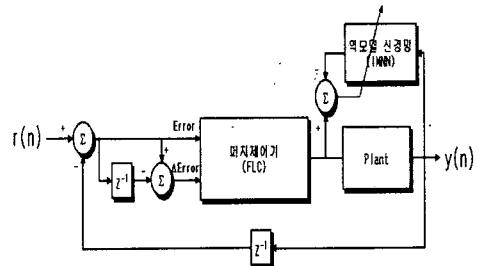


그림 4. 퍼지제어기를 사용한 플랜트 역모델 학습 구조.

Fig. 4. The learning structure of inverse model by the fuzzy controller.

신경망이 플랜트를 제어하기 위해서는 먼저 플랜트의 역모델을 학습하여야 한다. 이를 위해 퍼지제어기를 사용하였으며 학습구조는 그림 4와 같다. 학습이 완료

된 신경망은 퍼지보상기를 갖는 신경망제어기로 동작하게 된다.

## 2.2 퍼지보상기

그림 5는 퍼지보상기의 구조를 나타낸다. 설정치가 변경된 후, 첫 번째 최대 오버슈트를 전후로 하여 제어 규칙과 소속함수가 변경되도록 하였다.

첫 번째 오버슈트가 발생하기 전에는 제어규칙1(표 1 참조)을 사용하여 오버슈트를 감소시킨다. 이때, 사용한 오차  $e_{flc}$ 와 오차변화율  $\Delta e_{flc}$ , 플랜트 입력  $u_{flc}$ 의 소속함수는 그림 6과 같다.

첫 번째 오버슈트 발생 이후에는 제어규칙2(표 2 참조)를 사용하게 되고, 이때 오차  $e_{flc}$ 와 오차변화율  $\Delta e_{flc}$ 의 소속함수는 그림 7과 같으며 플랜트 입력  $u_{flc}$ 의 소속함수는 그림 6과 같다. 제어규칙2는 설정치에 더 빨리 도달할 수 있도록 해준다. 그리고 제어규칙 전환시 불연속은 신경망제어기에 의해서 보상된다.

퍼지추론의 계산용이를 위해 가우스 함수를 소속함수로 사용하였으며 간략화된 무게중심법을 사용하여 비퍼지화시켰다.

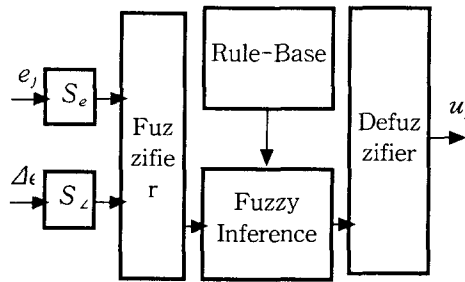


그림 5. 퍼지보상기 구조.

Fig. 5. The structure of fuzzy compensator.

표 1. 제어규칙 1.

Table 1. The control rule 1.

E	NB	NMB	NM	NMS	NS	NVS	ZE	PVS	PS	PMS	PM	PMB	PB
NB	NB	NB	NB	NB	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE
NMB	NB	NB	NB	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE	PVS
NM	NB	NB	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE	PS	PS
NMS	NB	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE	PVS	PS	PS
NS	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE	ZE	PS	PS	PS
NVS	NB	NB	NVS	NVS	NVS	NVS	ZE	ZE	PVS	PVS	PVS	PS	PS
ZE	NS	NS	NVS	NVS	NVS	ZE	ZE	ZE	PVS	PVS	PVS	PS	PS
PVS	NS	NS	NVS	NVS	NVS	ZE	ZE	PVS	PVS	PVS	PVS	PB	PB
PS	NS	NS	ZE	NVS	ZE	PVS	PS	PMS	PM	PMB	PB	PB	PB
PMS	NS	NS	ZE	ZE	PVS	PS	PMS	PM	PMB	PB	PB	PB	PB
PM	NS	ZE	ZE	PVS	PS	PMS	PM	PMB	PB	PB	PB	PB	PB
PMB	NVS	ZE	PVS	PS	PMS	PM	PMB	PB	PB	PB	PB	PB	PB
PB	ZE	PVS	PS	PMS	PM	PMB	PB	PB	PB	PB	PB	PB	PB

$$S_e = S_{\Delta e} = \frac{1}{R - R_{old}}, \quad (8)$$

$R$ : 설정치,  $R_{old}$ : 이전 설정치

$S_e$ :  $e_{flc}$ 을 [-1, 1]로 정규화하기 위한 스케일링 계수

$S_{\Delta e}$ :  $\Delta e_{flc}$ 을 [-1, 1]로 정규화하기 위한 스케일링 계수

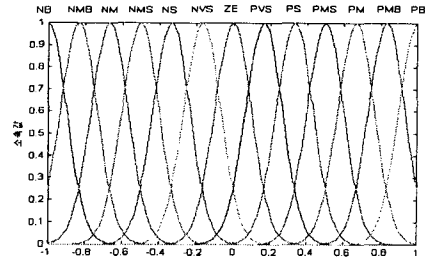


그림 6. 첫 번째 오버슈트에 도달하기 전에 사용된 오차( $e_{flc}$ )와 오차변화율( $\Delta e_{flc}$ ) 소속함수와 플랜트 입력( $u_{flc}$ ) 소속함수.

Fig. 6. Before arrive at the first overshoot, used to error ( $e_{flc}$ ) and change rate of error membership function ( $\Delta e_{flc}$ ), plant input membership function ( $u_{flc}$ ).

표 2. 제어규칙 2

Table 2. The control rule 2

E	NB	NMB	NM	NMS	NS	NVS	ZE	PVS	PS	PMS	PM	PMB	PB
NB	NB	NB	NB	NB	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE
NMB	NB	NB	NB	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE	PVS
NM	NB	NB	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE	PVS	PS
NMS	NB	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE	PVS	PS	PMS
NS	NB	NB	NMB	NM	NM	NM	NM	ZE	PM	PM	PS	PMS	PM
NVS	NB	NB	NMB	NM	NM	NVS	NM	ZE	PM	PM	PMS	PM	PMB
ZE	NB	NMB	NM	NMS	NM	NVS	ZE	PVS	PM	PM	PM	PMB	PB
PVS	NMB	NM	NMS	NS	NM	ZE	PM	PVS	PM	PM	PMB	PB	PB
PS	NM	NMS	NS	NVS	ZE	PM	PM	PM	PM	PMB	PB	PB	PB
PMS	NMS	NS	NVS	ZE	PVS	PS	PMS	PM	PMB	PB	PB	PB	PB
PM	NS	NVS	ZE	PVS	PS	PMS	PM	PMB	PB	PB	PB	PB	PB
PMB	NVS	ZE	PVS	PS	PMS	PM	PMB	PB	PB	PB	PB	PB	PB
PB	ZE	PVS	PS	PMS	PM	PMB	PB	PB	PB	PB	PB	PB	PB

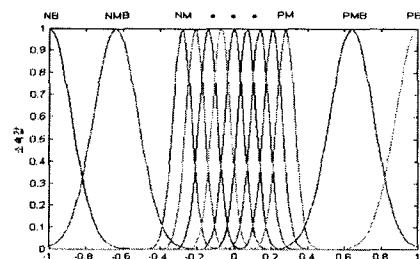


그림 7. 첫 번째 오버슈트 이후의 오차( $e_{flc}$ )와 오차변화율( $\Delta e_{flc}$ )의 소속함수.

Fig. 7. After arrive at the first overshoot, used to error ( $e_{flc}$ ) and change rate of error membership function ( $\Delta e_{flc}$ ).

### III. 모의실험

본 논문에서 제안한 제어기의 응답특성을 살펴보기 위해 다음의 2차 플랜트 전달함수에 적용하여 모의 실험하였다.

$$G(s) = \frac{1.2033 \times 10^8}{s^2 + 2.9627s + 6.5775 \times 10^6} \quad (9)$$

그림 8, 그림 9, 그림 10 그리고 그림 11은 TDNN 제어기의 모의 실험 결과이고, 그림 12, 그림 13, 그림 14 그리고 그림 15는 리커런트 TDNN 제어기의 모의 실험 결과이다.

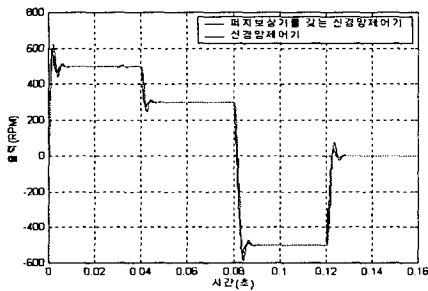


그림 8. TDNN 신경망제어기와 퍼지보상기를 갖는 TDNN 신경망제어기의 응답.  
Fig. 8. The response of the TDNN neural network controller and the TDNN neural network controller within fuzzy compensator.

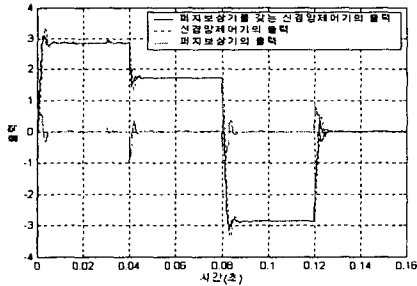


그림 9. 퍼지보상기를 갖는 TDNN 신경망 제어기, TDNN 신경망제어기, 퍼지보상기 시간 응답.  
Fig. 9. The time response of the TDNN neural network fuzzy compensator, the TDNN neural network controller and the fuzzy compensator.

설정치가 500rpm일 때, 두 제어기 모두 원하는 플랜트 출력에 도달하지만, TDNN 제어기에 비해 제안한 리커런트 TDNN 제어기가 4.887% 낮은 최대오버슈트와 3.7ms 빠른 정정시간을 가진다.(그림 8, 그림 12, 그리고 표 3 참조) 그림 9와 그림 12는 신경망제어기의 출력을 퍼지보상기가 보상하여 500rpm에 필요한 플랜트

트 입력으로 적절히 조절되고 있음을 나타내고 있다.

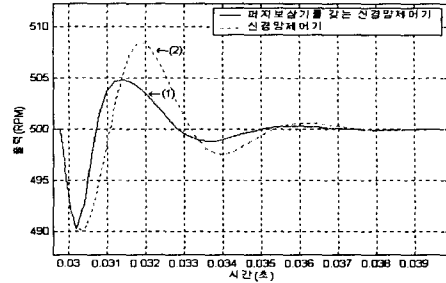


그림 10. 30ms에서 7rpm 외란을 가했을 경우의 퍼지보상기를 갖는 TDNN 신경망제어기(1), TDNN 신경망제어기(2)의 플랜트 출력 응답.  
Fig. 10. For a disturbance of 7rpm at 30ms, the response of the TDNN neural network controller within fuzzy compensator(1) and the TDNN neural network controller(2).

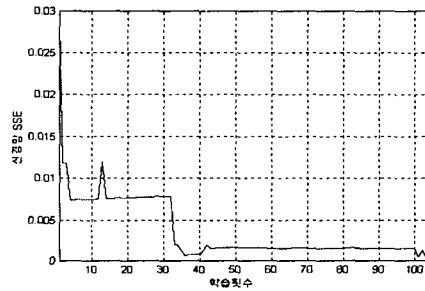


그림 11. TDNN 신경망 SSE 그래프.  
Fig. 11. The graphics of TDNN neural network SSE.(Sum of Squared Error)

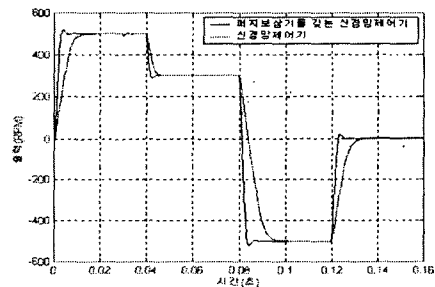


그림 12. 리커런트 TDNN 신경망제어기와 퍼지보상기를 갖는 리커런트 TDNN 신경망제어기의 응답.  
Fig. 12. The response of the Recurrent TDNN neural network controller and the Recurrent TDNN neural network controller within fuzzy compensator.

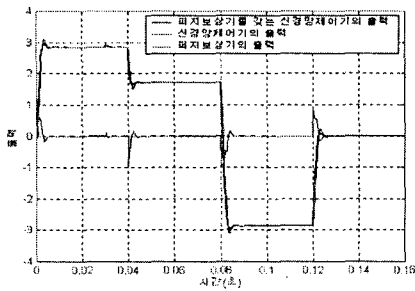


그림 13 퍼지보상기를 갖는 리커런트 TDNN 신경망 제어기, 리커런트 TDNN 신경망 제어기, 퍼지보상기 시간 응답.  
 Fig. 13. The time response of the Recurrent TDNN neural network controller within fuzzy compensator, the Recurrent TDNN neural network controller and the fuzzy compensator.

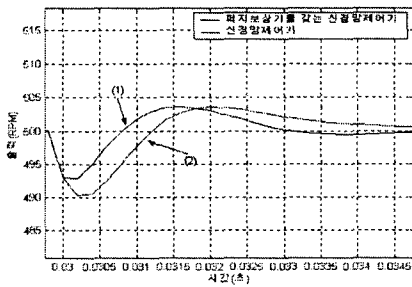


그림 14. 30ms에서 7rpm 외란을 가했을 경우의 퍼지보상기를 갖는 리커런트 TDNN 신경망제어기(1), 리커런트 TDNN 신경망제어기(2)의 플랜트 출력 응답.  
 Fig. 14. For a disturbance of 7rpm at 30ms, the response of the Recurrent TDNN neural network controller within fuzzy compensator(1) and the Recurrent TDNN neural network controller(2).

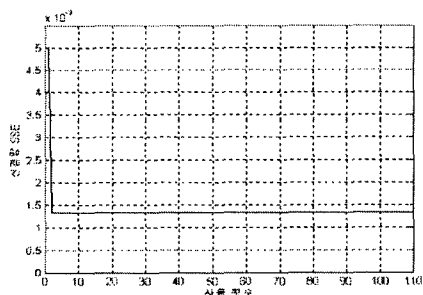


그림 15. 리커런트 TDNN 신경망 SSE 그래프.  
 Fig. 15. The graphics of Recurrent TDNN neural network SSE.(Sum of Squared Error)

표 3. 응답특성 비교.(500rpm)

Table 3. Comparison of a response characteristics. (500rpm)

	지연시간	상승시간	정정시간	최대 오버슈트	외란 인가시 수렴시간
TDNN 신경망제어기	1.7ms	1.81ms	6.7ms	8.94%	37ms
Recurrent TDNN 신경망제어기	1.6ms	2.036ms	3ms	4.053%	2.8ms

30ms에서 7rpm 외란을 가했을 때, 리커런트 TDNN 제어기가 TDNN 제어기에 보다 낮은 오버슈트와 3.7ms 빠른 수렴시간을 가졌다.(그림 8과 그림 12. 그리고 표 3 참조) 그리고, 그림 11에 나타낸 바와 같이 TDNN의 SSE는 초기엔 0.0285이지만 학습횟수가 110번일 땐 0.000376으로 감소되었다. 그리고 리커런트 TDNN의 SEE는 0.0055에서 시작해서 0.00133으로 수렴한다.

#### IV. 결론

본 논문에서는 플랜트를 완전하게 수학적으로 모델링하지 않고, 플랜트의 입출력만을 이용하여 제어기를 구성하였고, 신경망제어기의 성능을 향상시키기 위해 퍼지보상기를 이용하였다. 또 학습이 부족한 신경망제어기의 성능을 온라인으로 향상시키기 위해 플랜트의 역모델 신경망을 사용하였다.

플랜트의 역모델 신경망을 학습시킨 결과를 이용하여 주 신경망의 가중치를 변경시킴으로써 원하는 플랜트의 동적 특성을 얻도록 하였으며 2차 플랜트에 제안한 알고리즘을 적용하여 성능을 모의실험한 결과 TDNN 제어기에 비해 상승시간은 비록 길지만, 정상상태에 이르는 정정시간이 짧았고, 오버슈트가 정정시간 편차안에 들어간다. 최대오버슈트 역시 신경망제어기에 비해 낮았으며, 외란의 영향에 대해 제안한 제어기가 더욱 견실하였고 제어기의 학습횟수가 증가할수록, 즉 시간이 경과할수록 SSE가 감소하고 있음을 확인할 수 있었다.

본 논문에서 제안한 리커런트 TDNN 제어기가 TDNN 제어기보다 성능이 개선됨을 모의 실험을 통해 확인할 수 있었다.

#### VI. 참고문헌

- [1] 이상배, "퍼지-뉴로제어 시스템", (주)교학사, pp.11 1~200, 1999
- [2] Peter Vas, "Artificial-Intelligence-Based Electrical Machines and Drives", Oxford University Press, Inc., pp.77~234, 1999
- [3] E.H.Mamdani, "Twenty years of fuzzy control :

- Experiences gained and lessons learnt", proc. 2nd IEEE Int. Conf. fuzzy systems, San Francisco, CA, pp339-344, 1993
- [4] 김용태, "신경망과 퍼지보상기를 이용한 유압 서보 모터의 속도제어기 설계", 석사학위논문, 2000
- [5] Lefteri.H.Tsoukalas, Robert E.Uhrig, "Fuzzy and Neural Approaches in Engineering", John Wiley & Sons, Inc., pp.13~405, 1997
- [6] Simon Haykin, "Neural Networks, A comprehensive foundation ,Second Edition", Prentice Hall International, Inc., pp.1~255, pp.664~789, 1999
- [7] Michael A.Arbib, "The Handbook of Brain Theory and Neural Networks", The MIT Press, pp.19~20, 1995
- [8] Madan M.Gupta, Dandina H.Rao, "Dynamic Neural Processor and Its Applications to Robotics and Control: Intelligent Control Systems(Theory and Applications) Edited by Madan M.Gupta, Naresh K.Sinha", IEEE Press, pp.515~545, 1996
- [9] T. Takagi and M. Sugeno, "Derivation of fuzzy control rules from human operator's control actions", in Proc. of the IFAC Symp. on Fuzzy Information, Knowledge Representation and Decision Analysis, Marseille, France, 1983.
- [10] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller", Int. J. Man-Machine Studies, Vol. 7, No. 1, 1975.