

유압서보모터를 위한 퍼지보상기를 갖는 신경망제어기 설계 및 구현

김 용 태*, 이 상 윤**, 신 위 재**, 유 관 식**

* (주)에이엠타, **경남대학교 전자공학과

Design and Implementation of Neural Network Controller with a Fuzzy Compensator for Hydraulic Servo-Motor

Yong-Tae Kim*, Sang-Yun Lee**, Wee-Jae Shin**, Khwan-Sick Yoo**

*AMT CO., LTD, **Dept. of Electronic Engineering Kyungnam University

email: cwee@kyungnam.ac.kr

요약

본 논문에서는 신경망제어기의 출력을 보상하는 퍼지보상기를 갖는 신경망제어기에 관하여 제안하였다. 학습이 완료된 신경망제어기를 사용하더라도 예상치 못한 외란으로 인해 플랜트의 출력이 좋지 못한 경우가 있는데, 이것을 적절하게 조절해 주기 위해 퍼지보상기를 사용하여 원하는 결과를 얻을 수 있도록 하였다. 그리고 플랜트의 역모델 신경망을 학습시킨 결과를 이용하여 주 신경망의 가중치를 변경시킴으로써 원하는 플랜트의 동적 특성을 얻게 된다. 제안한 제어기의 성능을 확인하기 위해 유압 서보시스템을 대상으로 DSP 프로세서를 사용하여 구현한 후 실험결과를 관찰하였다.

ABSTRACT

In this paper, we proposed a neural network controller with a fuzzy compensator which compensate a output of neural network controller.

Even if learn by neural network controller, it can occur a 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 an inverse model neural network of plant, so a expected dynamic characteristics of plant can be got. In order to confirm a performance of the proposed controller, we implemented the controller using the DSP processor and applied in a hydraulic servo system. And then we observed an experimental results.

I. 서론

신경망은 비선형시스템을 모델링하고 근사화시킬 수 있으며, 비선형 시그모이드 활성화함수를 사용하여 전통적 제어법으로 접근하지 못하는 난해한 비선형 제어문제를 해결하기 위한 비선형 사상능력을 제공해 준다. 하지만, 학습시간이 많이 소요되고, 학습에 필요한 플랜트의 입·출력데이터를 구하기 어렵다.^{[1][2][3]} 또한, 학습이 완료되어진 신경망을 사용하더라도 예상치 못한 외란으로 인해 플랜트의 출력이 좋지 못하다.

본 논문에서는 이와 같이 학습된 신경망제어기의 플

랜트 출력이 좋지 못한 결과를 가져올 때, 이를 퍼지보상기를 사용하여 원하는 결과를 얻을 수 있도록 하는 신경망제어기를 설계하고 DSP 프로세서를 사용하여 구현한 후 비선형 플랜트인 유압 서보시스템에 적용하여 그 성능을 확인하였다.

II. 신경망제어기 설계

퍼지보상기를 갖는 신경망제어기는 주 제어기인 신경망제어기와 신경망제어기의 출력을 보상시켜주는 퍼지보상기, 플랜트의 역모델을 학습하는 역모델 신경망(Inverse Model Neural Network: IMNN)으로 구성되고, [그림 1]에 나타내었다.

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 $u_{flc}(n)$ 을 계산한다.

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

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

신경망제어기의 출력 $u_{nnc}(n)$ 와 퍼지보상기의 출력 $u_{flc}(n)$ 의 합이 플랜트의 입력 $u_p(n)$ 가 된다.

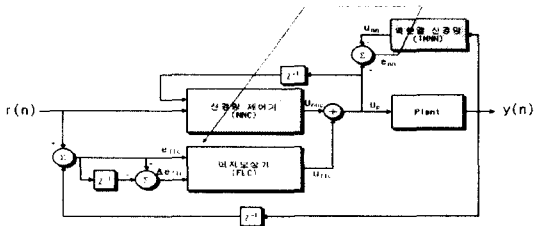


그림 1. 퍼지보상기를 갖는 신경망제어기 구조

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

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, 신경망제어기의 가중치를 역모델 신경망의 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)을 사용하였으며, 역모델 신경망에는 NARX(nonlinear autoregressive with exogenous inputs) 신경망을 사용하였다.^{[1][4]} 두 신경망 모두 시간지연을 가지기 때문에, 신경망 구조를 차분방정식 또는 순환방정식으로 기술할 수 있다.[3] 따라서, 시간에 따라 변하는 비선형 동적시스템을 모델링하기에 유리하다. 그리고, 플랜트 역모델을 온-라인으로 학습시킬 수 있기 때문에, 학습이 부족한 신경망제어기의 성능을 온-라인으로 보완시킬 수 있다.

학습알고리즘은 일반적으로 신경망을 학습시키는데 사용하는 오류 역전파 학습 알고리즘을 사용하였다. 지역 극소점에 빠지는 것을 방지하고 빠른 학습을 위하여 모멘텀법을 사용하였으며 가중치 조절식은 다음과 같다.^{[1][3]}

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

$$\Delta W_{jk}(n+1) = W_{jk}(n)x(n) + \alpha \Delta W_{jk}(n) \quad (5)$$

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

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

η : 학습률, α : 모멘텀 상수, $x(n)$: 입력층 뉴런값

$h(n)$: 은닉층 뉴런값, $y(n)$: 출력층 뉴런값

$W_{ij}(n)$: 입력층과 은닉층사이의 가중치

$W_{jk}(n)$: 은닉층과 출력층사이의 가중치

$$\Delta W = W_{new} - W_{old}$$

본 논문에서는 학습률 η , 모멘텀 상수 α 를 각각 0.4

와 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.

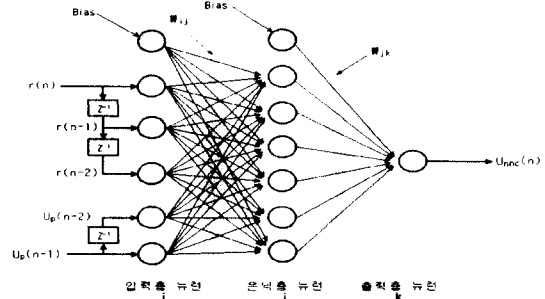


그림 2. 신경망제어기로 사용된 시간지연신경망(TDNN)

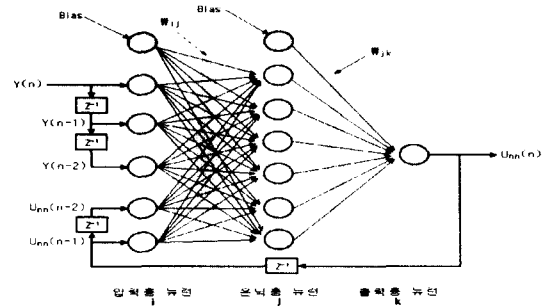


그림 3. 역모델 신경망에 사용된 NARX 신경망

신경망이 플랜트를 제어하기 위해서는 먼저 플랜트의 역모델을 학습하여야 한다. 이를 위해 퍼지제어기를 사용하였으며 학습구조는 [그림 4]와 같다. 학습이 완료된 신경망은 퍼지보상기를 갖는 신경망제어기로 동작하게 된다.

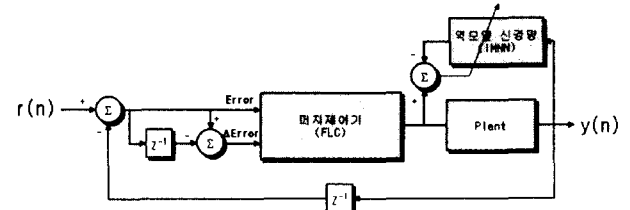


그림 4. 퍼지제어기를 사용한 플랜트 역모델 학습

2.2 퍼지보상기^[5]

$$S_e = S_{De} = \frac{1}{R - R_{old}}$$

R : 설정치, R_{old} : 이전 설정치

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

S_{De} : Δe_{flc} 을 [-1, 1]로 정규화하기 위한 스케일링 계수

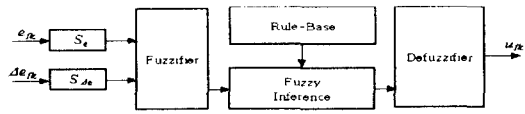


그림 5. 퍼지보상기 구조

[그림 5]은 퍼지보상기의 구조를 나타낸다. 첫 번째 오버슈트를 전후로 하여 제어규칙과 소속함수가 변경되도록 하였다.

첫 번째 오버슈트가 발생하기 전에는 제어규칙1을 사용하여 오버슈트를 감소시킨다. 이때, 사용한 오차 e_{flc} 와 오차변화율 Δe_{flc} , 플랜트 입력 u_{flc} 의 소속함수는 [그림 7]와 같다.

퍼지추론의 계산용이를 위해 가우스 함수를 소속함수로 사용하였으며 간략화된 무게중심법을 사용하여 비퍼지화시켰다.([그림 6] 참조)

첫 번째 오버슈트 발생 이후에는 제어규칙2를 사용하게 되고, 이때 오차 e_{flc} 와 오차변화율 Δe_{flc} 의 소속함수는 [그림 8]과 같으며 플랜트 입력 u_{flc} 의 소속함수는 [그림 7]와 같다. 제어규칙2는 설정치에 더 빨리 도달할 수 있도록 해준다.

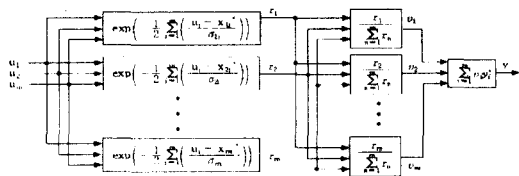


그림 6. 퍼지추론과 비퍼지화

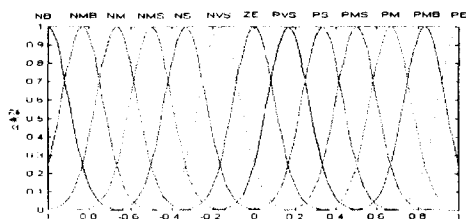


그림 7. 첫 번째 오버슈트에 도달하기 전, e_{flc} , Δe_{flc} , 소속함수와 u_{flc} 소속함수

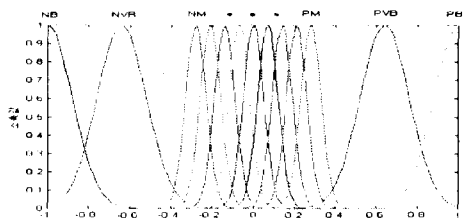


그림 8. 첫 번째 오버슈트 이후, e_{flc} , Δe_{flc} , 소속함수

표 1. 제어규칙1

E														
ΔE	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	NB	NMB	NM	NMS	NS	NVS	ZE	PVS
NM	NB	NB	NB	NB	NB	NB	NMB	NM	NMS	NS	NVS	ZE	PS	PS
NMS	NB	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	PM	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	

표 2. 제어규칙2

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

III. 실험 결과

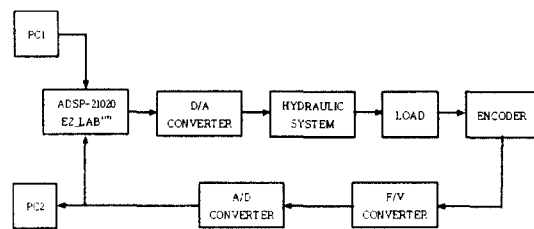


그림 9. 실험장치의 구성도

[그림 9]의 블록 선도는 유압 서보모터시스템의 구성을 나타내었다.^[6] [그림 9]에서, 퍼지보상기를 갖는 신경망 제어기의 제어 알고리즘을 수행하는 부분은 ADSP-21020 EZ-Lab™ DSP 보드이고, 이 보드의 제어는 RS-232 통신 방식을 사용하여 PC1에서 이루어진다. PC1은 ADSP-21020 계열 개발 소프트웨어를 사용하여 C 언어로 작성한 뒤 제안한 알고리즘을 DSP 어셈블리 언어로 번역하고, 실행파일을 생성시켜 ADSP-21020 EZ-Lab™ DSP 보드로 다운로드(Down loading)하여 알고리즘을 수행한다. 유압 서보시스템은 유압모터와 직접구동 밸브(Direct Drive Valve) 그리고 관성부하로 이루어진다. 유압 서보시스템에서 제안한 신경망제어기의 제어 입력신호를 D/A 변환기를 사용하여 아날로그 전압

으로 바꾼 다음 직접구동 밸브에 인가한다. 직접구동 밸브의 구동신호(제어입력)는 $\pm 10V$ 로 ADSP-21020 EZ-Lab™ DSP 보드에 D/A 변환기(AD669 : 16bit)를 연결하여 생성하며 직접구동 밸브의 A, B 포트(Port)는 유압모터의 포트에 연결되고, 유압모터의 축은 커플링을 통하여 관성부하를 연결하는 축에 결합된다. 밸브에 의해 제어된 유량은 유압모터로 공급되고, 공급된 유량에 비례해서 회전력을 발생시킨다. 또한 관성부하를 고정하는 축은 다시 커플링을 통하여 증분 광부호기로 연결된다. 증분 광부호기는 유압모터의 회전수를 감지하는 부분으로 F/V 변환기를 사용하여 유압모터의 회전수를 전압신호로 변환하고, 변환된 전압신호는 A/D 변환기를 통과하여 ADSP-21020 EZ-Lab™ DSP 보드로 연결된다. 또한 PC2의 A/D 변환 카드 (Lab-PC-1200/AI)로 전달되며, PC2는 실험 결과를 기록하는 역할을 담당한다. 유압 서보시스템 실험장치는 서보밸브를 100% 열고, 유압펌프의 압력을 최대로 하면, 약 250[rpm]으로 모터가 회전하지만, 실험은 설정치가 ± 150 [rpm]일때 공급 유압 60[kg/cm²]으로 하고, 설정치가 ± 200 [rpm]일 때 80[kg/cm²]의 유압을 유지하여 수행하였다. 이상의 실험 결과는 [표 3]에서 비교 검토하였다.

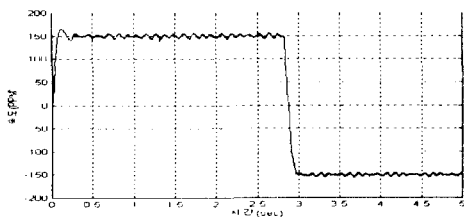


그림 10. 퍼지보상기를 갖는 신경망제어기의 응답 (기준치: ± 150 rpm, Oil 압력: 60[kg/cm²], 관성부하: 79.882[kg/cm²])

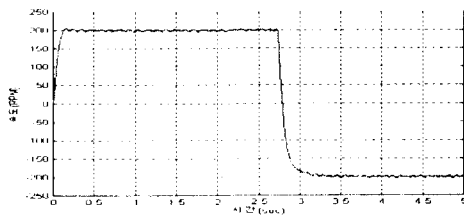


그림 11. 퍼지보상기를 갖는 신경망제어기의 응답 (기준치: ± 200 rpm, Oil 압력: 80[kg/cm²], 관성부하: 79.882[kg/cm²])

표 3. 실험결과

	지연시간	상승시간	정정시간	최대 오버슈트
± 150 rpm	26ms	49.6ms	148ms	9.7 %
± 200 rpm	39.4ms	80.2ms	108ms	3.1 %

실험 결과를 살펴보면 ± 150 [rpm]보다 ± 200 [rpm]일 때 더 나은 결과를 얻을 수 있다. 이는 유압 서보시스

템의 역모델을 ± 200 [rpm]로 학습시켰기 때문이다.

[그림 10]과 [그림 11]을 살펴보면 정상상태에서 리플이 주기적으로 발생한다는 것을 볼 수 있다. 이는 부하와 유압 모터 사이에 연결된 스프링 커플링의 관성작용으로 인해 나타나는 것이다.

리플이 주기적으로 좀 더 크게 나타나는 것은 스프링 커플러에 의한 것이고, 큰 리플 속에 포함된 작은 리플은 제어기에서 디지털 회로가 동작할 때 나타나는 스위칭 잡음으로서 ADSP-21020 EZ-LAB™ DSP 보드가 제어 알고리즘을 수행할 때 DSP 프로세서에 의해서 유도되는 잡음이다. 필요한 노이즈 대책을 적용하여 잡음을 많이 제거하였기 때문에 측정 데이터를 확대하여 보면 A/D 변환시 발생하는 양자화 잡음으로 생각해도 무방하다.

IV. 결론

일반적으로 제어기 설계 방법은 먼저 제어하고자 하는 제어대상 즉 플랜트에 대한 자료를 수집하여 모델링한 다음 제어기를 설계한다. 그러나 본 논문에서는 플랜트를 완전하게 수학적으로 모델링하지 않고, 플랜트의 입출력만을 이용하여 제어기를 설계하였다.

본 논문은 속도제어기로서 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 변경시킴으로서 원하는 플랜트의 동적 특성을 얻도록 하였다.

그리고, 비선형 유압 서보모터의 실시간 제어를 위해 DSP 프로세서를 이용하여 제어기를 구현하였으며 퍼지보상기를 갖는 신경망제어기의 성능을 실험을 통해 확인할 수 있었다.

V. 참고문헌

- [1] Lefteri H. Tsoukalas, Robert E.Uhrig, "Fuzzy and Neural Approaches in Engineering", John Wiley & Sons, Inc. pp.238-250, p333-382, 1997
- [2] Peter Vas, "Artificial-Intelligence-based Electric Machines and Drives", Oxford University Press, pp.152 - 170, 1999
- [3] 이상배, "퍼지-뉴로제어 시스템", (주)교학사, pp.153-194, 1999
- [4] Simon Haykin, "Neural Networks", Prentice-Hall, Inc. pp.732-747, 1999
- [5] Ronald R.Yager, Dimitar P.Filev, "Essentials of Fuzzy Modeling and Control", John Wiley & Sons, Inc. pp.109-202, 1994
- [6] 이상운, "지능구조를 갖는 폐환제어기 설계에 관한 연구", 박사학위논문, 경남대학교 대학원 전자공학과, 1998