

RTDNN과 FLC를 사용한 신경망제어기 설계

Design of Neural Network Controller Using RTDNN and FLC

신 위 제*

Wee-Jae Shin*

요 약

본 논문에서는 RTDNN 과 FLC를 이용해서 주신경망을 보상하는 제어시스템을 제안한다. 주신경망이 학습을 완료한 후 외란이나 부하변동이 생겨 오브 슈트 내지는 언더 슈트를 나타낼 때 적절히 조정하기 위해 퍼지 보상기를 사용하여 원하는 결과를 얻을 수 있도록 하였다. 그리고 제어대상의 역모델 신경망에서 학습시킨 결과를 이용하여 주신경망의 가중치를 변경시킴으로서 제어대상의 원하는 동적 특성을 얻게 된다. 모의 실험 결과 제안한 신경망 제어기의 양호한 응답 특성을 확인 할 수 있다.

Abstract

In this paper, We propose a control system which compensate a output of a main Neual Network using a RTDNN(Recurrent Time Delayed Neural Network) with a FLC(Fuzzy Logic Controller)After a learn of main neural network , it can occur a Over shoot or Under shoot from a disturbance or a load variations. 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. We can confirm good response characteristics of proposed neural network controller by the results of simulation

keywords : neural network controller, tdnn, rtdnn ,FLC, scale coefficient

I. 서론

신경망은 비선형 시그모이드 활성화함수를 사용하여 전통적 제어방법으로 접근하지 못하는 난해한 비선형 제어 문제를 해결하기 위한 비선형 사상능력을 제공해 준다.[4,6] 하지만, 학습시간이 많이 소요되고, 학습에 필요한 제어대상의 입출력 데이터를 필요로 한다. 또한, 학습이 완료되어진 신경망을 사용하더라도 예상치 못한 외란으로 인해 제어대상의 출력이 좋지 못한 경우도 있다. 퍼지제어기는 제어대상에 대한 작업자의 경험이나 전문가의 언어적인 표현을 쉽게 제어기와 결합시킬 수 있는 장점을 지니고 있다. 또한 제어대상의 파라미터 변동이나 외부 외란에 크게 영향을 받지 않는 강인한 제어기의 특성도 지니고 있다.[1]

본 논문에서는 RTDNN에[2][7] 퍼지 보상기를 사용하여 스케일계수를 자동 조정함으로써 주신경망제어기의 가중치가 자동 조정되어 제어성능이 개선됨을 보였다.

II. TDNN 의 구조

신경망 제어기에는 시간지연 신경망(Time Delayed Neural Network: TDNN)[5][6]을 사용하였으며, 역모델 신경망은 RTDNN을 이용하여 구성하였다. 사용하는 신경망은 시간지연 기능을 갖기 때문에, 신경망 구조를 차분방정식 또는 순환방정식으로 기술할 수 있다. 따라서, 시간에 따라 변하는 비선형 동적 시스템을 모델링하기에 유리하다.[4][8] 그리고 제어대상의 역모델을 온-라인으로 학습시킬 수 있기 때문에, 학습이 부족한 신경망제어기의 성능을 온-라인으로 보완시킬 수 있다.

* 경남대학교 전자공학과

투고 일자 : 2012 10. 19 수정완료일자 : 2012. 11. 1

게재확정일자 : 2012. 11. 3

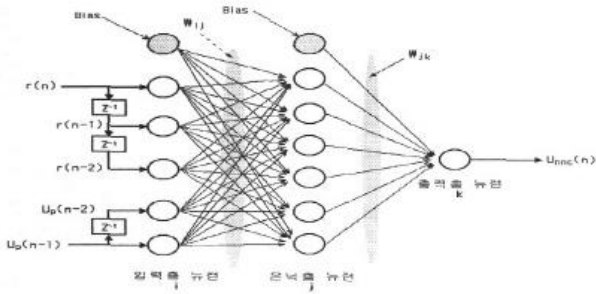


그림1. TDNN 신경망제어기

Fig.1. TDNN in Neural Network Controller

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.

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

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

$$W_{ij}(n+1) = W_{ij}(n) + \Delta W_{ij}(n+1) \quad (3)$$

$$W_{jk}(n+1) = W_{jk}(n) + \Delta W_{jk}(n+1) \quad (4)$$

III. 퍼지제어기를 사용한 역모델 신경망의 구조와 학습

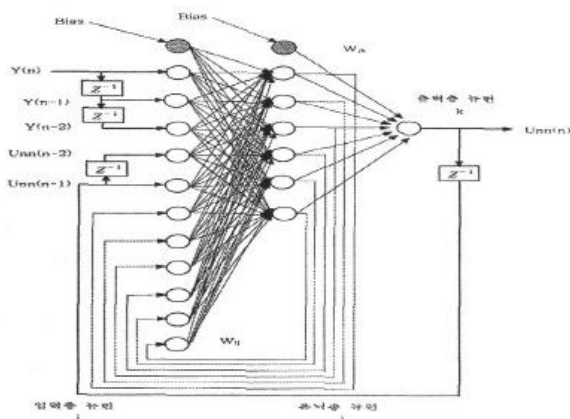


그림2. 역모델 신경망 제어기

Fig.2. Inverse Neural Network

본 논문에서는 학습률 η , 모멘텀 상수 α 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 I

번째 입력층과 j번째 은닉층사이의 가중치, $W_{jk}(n)$ 는 i번째 은닉층과 k번째 출력층사이의 가중치를 나타낸다.

역모델 신경망은 그림 2의 RTDNN을 사용하였으며, 구조상으로 신경망제어기와 차이점은 없다. 역모델 신경망은 신경망 제어기에서 기준 입력 $r(n)$, 기준입력의 1차 지연 $r(n-1)$, 기준입력의 2차 지연 $r(n-2)$ 그리고 과거의 제어대상의 입력의 1차 지연 $u_p(n-1)$, 과거의 제어대상의 입력의 2차 지연 $u_p(n-2)$ 대신에 현재의 제어대상 출력 $y(n)$, 제어대상 출력의 1차 지연 $y(n-1)$, 제어대상 출력의 2차 지연 $y(n-2)$ 그리고 신경망제어기의 제어출력의 1차 지연 $u_{nnc}(n-1)$, 신경망 제어기의 제어출력의 2차 지연 $u_{nnc}(n-2)$ 을 이용하여 신경망 제어기의 제어출력 $u_{nnc}(n)$ 을 학습한다.

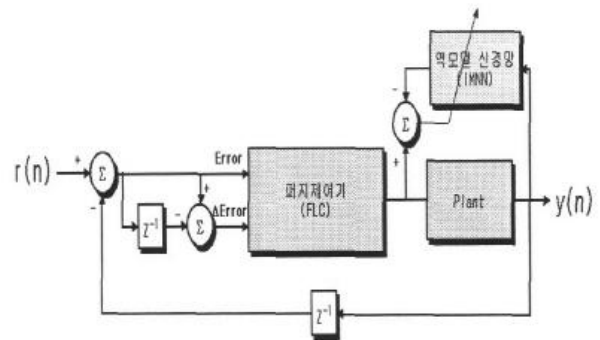


그림3. FLC를 이용한 플랜트역모델 학습구조

Fig.3. Learning Structure of The Plant Inverse Model Using FLC

신경망 제어기가 제어대상을 제어하기 위해서는 먼저 제어대상의 역모델 신경망을 학습 시켜야 한다. 퍼지제어기를 사용하여 역모델 신경망을 학습하기 위한 구조는 그림 3과 같다. 초기 역모델 신경망을 학습시키기 위해 주 제어기는 퍼지 제어기가 담당한다. 그리고 퍼지제어기의 제어 알고리즘을 역모델 신경망에 학습시킨다. 학습이 완료된 후 역모델 신경망은 학습한 가중치를 신경망제어기에 전달한다. 가중치가 수정된 신경망 제어기는 그림 4와 같은 퍼지보상기를 갖는 신경망제어기로 동작하게 된다.

IV. FLC 구조와 추론

그림4는 퍼지보상기의 구조를 나타낸다. 퍼지보상기는 퍼지화기, 퍼지추론, 비퍼지화기로 구성되며 퍼지추론 메카니즘을 기반으로 한다.

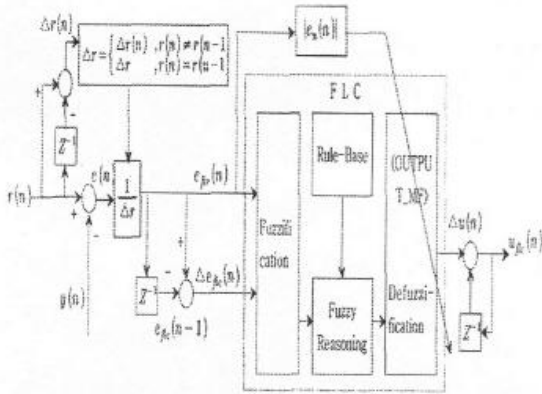


그림4. FLC구조
Fig.4. FLC Structure

사용된 퍼지 보상기에는 다음과 같은 퍼지추론과 비 퍼지화 과정이 사용되었다.

IF e_{flc} is B_{i1} AND Δe_{flc} is B_{i2} THEN V is D_i , $i=(1, 2, \dots, n)$ (5)

오차 $e_{flc}(k)$ 와 오차변화량 $\Delta e_{flc}(k)$ 의 정규화는 두 가지 경우로 나누어 생각할 수 있다. 첫 번째 설정치가 바뀌지 않고 과거의 설정치 $r(k-1)$ 가 현재의 설정치 $r(k)$ 를 유지할 때 과거의 Δr 을 그대로 유지한다. 두 번째는 설정치가 변화하여 과거의 설정치 $r(k-1)$ 와 현재의 설정치가 같지 않을 때는 Δr 을 $r(k) - r(k-1)$ 으로 바꾸어준다. 식(6)과 같이 설정치의 변화량 Δr 을 계산한 다음 식(7)에 적용한다. 식(8)은 오차 $e(k)$ 를 설정치의 변화량으로 정규화 한다.

$$\Delta r = \begin{cases} r(k) - r(k-1), & r(k) \neq r(k-1) \\ \Delta r, & r(k) = r(k-1) \end{cases} \quad (6)$$

$$e(k) = r(k) - y(k) \quad (7)$$

$$e_{flc}(k) = \frac{r(k) - y(k)}{\Delta r} \quad (8)$$

$$\Delta e_{flc}(k) = e_{flc}(k) - e_{flc}(k-1) \quad (9)$$

$\Delta e(k)$ 는 식(9)으로 계산된다.

간략화된 퍼지 추론법에 따른 퍼지모델의 크리스프 출력은 COA(Center of Area)를 사용한다.

$$du = \frac{\sum_{i=1}^m \tau_i du_i^*}{\sum_{i=1}^m \tau_i} \quad (10)$$

여기서, du_i^* 는 결론부 퍼지집합 D_i 의 비퍼지화된 값이다.

τ_i 는 주어진 cr크리스프 입력값 e_{flc} , Δe_{flc} 에 대한 규칙의 적합도이다.

$$\tau_i = B_{i1}(e_{flc}) \wedge B_{i2}(\Delta e_{flc}) \quad (11)$$

간략화된 퍼지 추론방법은 개별적인 규칙들의 적합도 합수 τ_i 로 퍼지모델의 cr크리스프 출력은 해석적 수식으로 된다. 그러나 이것은 크리스프 입력값 e_{flc} 과 Δe_{flc} 으로 이루어진 함수는 아니다. 식(11)에서 해석적 수식을 얻기 위해 min 연산자를 product로 교체하고 전체 퍼지집합은 파라미터가 x_i^* 와 σ_i^* 를 갖는 가우스 소속함수로 가정한다면 τ_i 는 다음과 같이 재정리하여 적을 수 있다.

$$\begin{aligned} \tau_i &= B_{i1}(e_{flc}) \cdot B_{i2}(\Delta e_{flc}) \\ &= \left[\exp\left(-\frac{1}{2} \left(\frac{e_{flc} - x_i^*}{\sigma_i}\right)^2\right) \cdot \exp\left(-\frac{1}{2} \left(\frac{\Delta e_{flc} - x_i^*}{\sigma_i}\right)^2\right) \right] \end{aligned} \quad (12)$$

여기서 x_i^* 와 σ_i^* 는 각각 가우스 함수의 중심과 폭이다. 식(12)를 식(10)에 대입하면 전제부와 결론부 퍼지집합의 파라미터 du_i^* , x_i^* 와 σ_i^* , $i=(1, m)$ 에 의해 결정되는 LM의 crisp출력은 다음과 같다.

$$\begin{aligned} \Delta u = du &= \frac{\sum_{i=1}^m \tau_i du_i^*}{\sum_{i=1}^m \tau_i} \\ &= \frac{\sum_{i=1}^m du_i^* \left[\exp\left(-\frac{1}{2} \left(\frac{e_{flc} - x_i^*}{\sigma_i}\right)^2\right) \cdot \exp\left(-\frac{1}{2} \left(\frac{\Delta e_{flc} - x_i^*}{\sigma_i}\right)^2\right) \right]}{\sum_{i=1}^m \left[\exp\left(-\frac{1}{2} \left(\frac{e_{flc} - x_i^*}{\sigma_i}\right)^2\right) \cdot \exp\left(-\frac{1}{2} \left(\frac{\Delta e_{flc} - x_i^*}{\sigma_i}\right)^2\right) \right]} \\ &= \sum_{i=1}^m v_i du_i^* \end{aligned} \quad (13)$$

여기서, v_i 는 다음과 같다.

$$\begin{aligned} v_i &= \frac{\tau_i}{\sum_{j=1}^m \tau_j} \\ &= \frac{\left[e\left(-\frac{1}{2} \left(\frac{e_{flc} - x_i^*}{\sigma_i}\right)^2\right) \cdot e\left(-\frac{1}{2} \left(\frac{\Delta e_{flc} - x_i^*}{\sigma_i}\right)^2\right) \right]}{\sum_{j=1}^m \left[e\left(-\frac{1}{2} \left(\frac{e_{flc} - x_j^*}{\sigma_j}\right)^2\right) \cdot e\left(-\frac{1}{2} \left(\frac{\Delta e_{flc} - x_j^*}{\sigma_j}\right)^2\right) \right]} \end{aligned} \quad (14)$$

여기서, x_{ij}^* , σ_{ij} , du_i^* 는 각각 가우스 함수의 중심, 가우스 함수의 폭, 결론부 퍼지집합의 비퍼지화된 값을 나타낸다. 퍼지 보상기의 증분 Δu 는 식(13)으로 주어지고, 퍼지 보상기의 출력 $u_{flc}(k)$ 은 식(15)와 같다.

$$u_{flc}(k) = u_{flc}(k-1) + \Delta u \quad (15)$$

또한, 퍼지 보상기에서 사용한 제어 규칙과 소속함수는 기준 입력이 변경된 후, 제어대상의 출력에서 첫 번째 최대 오버슈트를 전후로 하여 제어규칙과 소속함수가 변경되도록 하였다. 첫 번째 오버슈트가 발생하기 전에는 제어규칙 1을 사용하여 오버슈트를 감소시킨다. 이때, 사용한 오차

e_{flc} 와 오차변화율 Δe_{flc} 은 그림5를 사용하였다.

첫 번째 오버슈트 발생 이후에는 제어규칙2 를 사용하게 되고, 이때 오차 e_{flc} 와 오차변화율 Δe_{flc} 의 소속함수는 그림 5와 같다. 제어규칙2는 설정치에 더 빨리 도달할 수 있도록 해준다. 그리고 제어규칙 전환시 불연속은 신경망제어기에 의해서 제어된다.

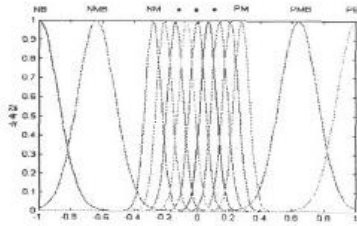


그림5. 첫번째 오버슈트 후의 오차 e_{flc} 와 오차변화율 Δe_{flc} 의 소속함수

Fig.5. Error e_{flc} After First Overshoot and The Membership Function of The Error Δe_{flc}

V. RTDNN과 퍼지보상기를 갖는 신경망제어기 설계

본 논문에서 제안하는 신경망 제어기는 그림 6 과 같이 주 제어기인 신경망제어기와 신경망제어기의 출력을 보상 시켜주는 퍼지보상기, 제어대상의 역모델을 학습하는 역모델 신경망(Inverse Model Neural Network: IMNN)으로 구성 된다. 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.

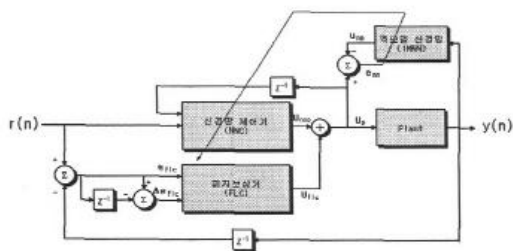


그림6. 퍼지 보상기를 갖는 신경망 제어기 구조
Fig.6. Structure of Neural Network Controller with The Fuzzy Compensator

$$e_{flc}(n) = r(n) - y(n-1) \quad (16)$$

$$\Delta e_{flc}(n) = e_{flc}(n) - e_{flc}(n-1) \quad (17)$$

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

$$u_p(n) = u_{nnc}(n) + u_{flc}(n) \quad (18)$$

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, 학습이 진행됨에 따라 신경망제어기의 성능을 향상시킬 수 있다.

VI. 모의실험

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

$$G(s) = \frac{1.2033 \times 10^8}{s^2 + 2.9627s + 6.5775 \times 10^6} \quad (19)$$

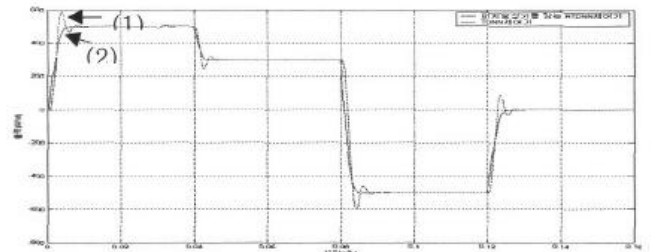


그림7. TDNN제어기(1)와 퍼지보상기를 가진 RTDNN응답(2)
Fig.7. Response of The RTDNN Controller with Fuzzy Compensator(2) and TDNN Controller (1)

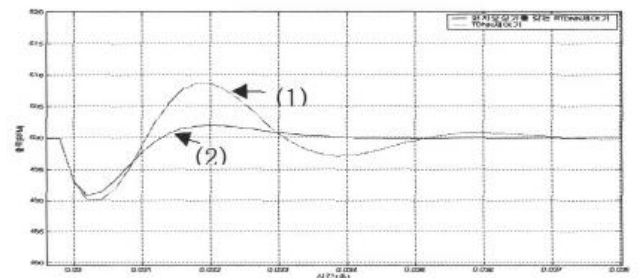


그림8. 30ms에서7rpm 외란을 가한 경우 퍼지보상기를 갖는 RTDNN 제어기의 플랜트응답(2)
Fig.8. Output Response of Plant of RTDNN Controller(2) With The Fuzzy compensator A Case of Driving Disturbance of 7rpm at 30ms

그림 7은 스케일 계수 자동조정하는 퍼지보상기를 갖는 리커런트 TDNN 제어기의 응답을 도시하였다. 오버슈트 없이 정상상태에 잘 도달하는 것을 관찰 할 수 있다. 그림 8은 외란 인가시의 응답을 나타낸다. TDNN 제어기와 스케일 계수 자동조정 퍼지 보상기를 갖는 RTDNN 제어기의 응답 특성을 표 1.에 비교하였다.

표1.성능지수(500rpm)
Table 1. Performance Index (500rpm)

	지연 시간	상승 시간	정정 시간	최대 오버슈트	외란 인가시 수렴시간
TDNN 제어기(1)	2.165ms	1.557ms	6.698ms	18.014%	36.8ms
제안한 제어기(2)	1.572ms	2.586ms	4.1285ms	0.0%	4.0ms

VII. 결론

본 논문에서는 플랜트를 완전하게 수학적 모델링하지 않고, 제어대상의 입력력만을 이용하여 제어기를 구성하였고, 신경망제어기의 성능을 향상시키기 위해 스케일 계수를 자동 조정하는 퍼지보상기를 이용하였다. 또 학습이 부족한 신경망제어기의 성능을 온라인으로 향상시키기 위해 제어대상의 역모델 신경망을 사용하였으며 제어대상의 역모델 신경망을 학습시킨 결과를 이용하여 주 신경망의 가중치를 변경시킴으로서 원하는 제어대상의 동적 특성을 얻도록 하였다. 본 논문에서 2차 제어대상을 부하로 하여 성능을 모의 실험한 결과 제안한 제어기가 오버슈트 없이 정상상태에 잘 도달하였음을 확인 하였다.

참고문헌

[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] Ronald R. Yager, Dimitar P. Filev, "Essentials of Fuzzy Modeling and Control", John Wiley & Sons, Inc., 1994.

[6] Michael A.Arbib, "The Handbook of Brain Theory and Neural Networks", The MIT Press, pp.19-20 (1995)

[1] 이상배, "퍼지-뉴로제어 시스템", (주)교학사, 1999.

[5] Simon Haykin, "Neural Networks, A comprehensive foundation ,Second Edition", Prentice Hall International, Inc., pp.1-255, pp.664-789(1999)

[2] Peter Vas, "Artificial-Intelligence-Based Electrical Machines and Drives", Oxford University Press, Inc., pp.77-234 (1999)

[7] 김용태, "신경망과 퍼지보상기를 이용한 유압 서보모터의 속도제어기 설계", 석사학위논문, (2000.)

[8] 신위재, "신경망-퍼지 하이브리드 제어기 설계" 신호처리.시스템 학회 논문지, 제3권1호, pp54-60 (2002)

[9] 정동하, 이상윤, 신위재, "정규화-가변 스케일 계수를 이용한 퍼지 제어기 설계 및 구현", 2003년도 추계학술대회 문집, 제4권 제2호, 한국 신호처리·시스템 학회 (2003)

[10] W.Pedrycz, M. Reformat. and C.W.Han, "Cascade architectures of fuzzy neural networks" Fuzzy Optimization and Decision making, Vol.3, no.1, pp 5-37 (2004.)

[11] C.C. Hua,Q.G. Wang,and X.P. Guan, "Adaptive fuzzy output-feedback controller design for nonlinear time-delay system with unknown control direction."IEEE Trans. Systems Man,and Cybernetics-partB,vol39, no 2 ,pp 363-374(2009)

[12] 신위재,정래원,"실내 이동로봇의 위치인식 및 속도제어에 관한 연구" 신호처리.시스템학회논문지,제11권1호,pp88-91(2010)



신 위 재 (Wee-Jae Shin)

1983년-현 경남대학교 전자공학과 교수
 2012년 - 현 한국신호처리시스템학회 회장
 1998년 일본 Kyushu 대학 객원교수
 1997년 대한전자공학회 학술위원
 1989년 동아대 전자공학과(공학박사)
 1975년 동아대 전자공학과 졸업
 *관심분야 : 시스템지능제어및신호처리