

신경망 가변구조제어기 설계

Neural Network Variable Structure Controller Design

박재삼*, 이진국**
(Jae-Sam Park, Chin-Gook Lhee)

요 약

본 논문에서는 가변구조제어기의 슬라이딩모드이득과 경계층두께(boundary layer thickness)를 신경망을 이용하여 계산하는 신경망 가변구조제어기를 제시한다. 제시된 방법은 신경망의 역전파오차 학습기능을 이용하여 슬라이딩모드이득과 경계층 범위를 계산할 수 있도록 신경망 제어기를 학습시킴으로써, 슬라이딩모드 제어법칙을 단순화 하고, 시스템 불확실성에 대하여 강인하며, 추적오차를 더욱 개선시킬 수 있다. 설계의 예와 시뮬레이션 결과를 통하여 제시된 방법에 대한 유용성을 보인다.

I. 서론

불확실성이 포함된 비선형 플랜트에 대한 강인한 제어기 설계의 한 방법으로 1950년대부터 가변구조제어(VSC; Variable Structure Control) 방법이 연구되어 왔다. 가변구조제어 이론은 불연속적인 제어입력을 이용하여 시스템의 구조를 의도적으로 가변시킴으로써 슬라이딩모드(sliding mode)라는 독특한 동특성으로 인하여 강인한 제어특성을 얻는 제어기법으로서 슬라이딩모드제어(SMC; Sliding Mode Control)는 가변구조제어의 한 형태이다. 가변구조시스템의 슬라이딩모드에 대한 연구는 1970년대부터 Utkin[8,9]등에 의하여 활발히 연구가 진행되어왔다.

이러한 가변구조의 슬라이딩모드 제어기법은, [8]에서 보는 바와 같이, 모델의 불확실성에 대한 강인성은 있지만, 스위칭평면에서의 불연속성으로 인하여 제어신호에 채터링(chattering)현상이 발생하여 적용에 한계가 있었다. 1980년대 중반, Slotine[6,7]은 스위칭평면에 경계층(boundary layer)을 도입하여 채터링을 없애는 방법을 보였다. 최근에는 적응제어, 최적제어, 또는 퍼지로지제어등과 결합한 형태의 제어 이론들이 제시되었다.

신경망(neural network)은 우수한 학습기능을 포함할 수 있으므로 최근 많은 연구가 발표되고 있다. 그러나 신경망의 가장 큰 단점은 네트워크가 어떻게 각각의 제어작업을 해결하는지 분명치 않다 점이며 따라서 제어시스템에 적용시, 제어시스템의 가장 기본인 안정도 판별이 어렵다는 불리한 점을 가지고 있다.

본 논문에서는 가변구조의 슬라이딩모드 제어기에 신경망을 결합한 새로운 신경망 슬라이딩모드 제어기를 설계한다. 이에 따라, 슬라이딩모드 제어법칙의 특성을 이용하여 전체 시스템의 안정도를 판별할 수 있으므로 제어기 설계가 용이하며, 신경망의 학습기능을 이용할 수 있으므로 제어대상의 비선형성이나 불확실성

* 박재삼 : 시립인천전문대학 전자과

** 이진국: 국립충주대학교 전기과

및 외란에 강인한 비선형 시스템에 적용이 용이한 제어기를 설계할 수 있다. 신경망의 학습절차는 많은 연구자들에 의하여 연구된 다양한 방법을 사용할 수 있지만 본 논문에서는 퍼지 오차 역전파 알고리즘(FEBP)을 사용한다[4].

본 논문은 2장, 슬라이딩모드 제어의 배경 설명; 3장, 신경망 슬라이딩모드 제어법칙 제시; 4장, 제어기의 구조; 5장, 적용 설계의 예 및 컴퓨터 시뮬레이션; 6장, 결론으로 구성된다.

II. 슬라이딩모드 제어의 배경

다음과 같은 SISO(single-input-single-output) 2차 동적시스템을 고려하여 보자.

$$\dot{\mathbf{x}} = f(\mathbf{x}) + b(\mathbf{x})u, \quad \mathbf{x} = [x \ \dot{x}]^T \quad (1)$$

여기서 u 는 제어입력이고, $\mathbf{x} = [x \ \dot{x}]^T$ 는 상태(state)벡터이며, $f(\mathbf{x})$ 를 \hat{f} 로 추정할 때, 다음과 같이 함수 F 내에 존재한다고 가정할 수 있으며,

$$|\hat{f}(\mathbf{x}) - f(\mathbf{x})| \leq F(\mathbf{x}) \quad (2)$$

$b(\mathbf{x})$ 를 \hat{b} 로 추정할 때 다음과 같이 나타낼 수 있다[7].

$$0 < b_{\min} \leq b(\mathbf{x}) \leq b_{\max} \quad (3)$$

$\mathbf{x}_d = [x_d \ \dot{x}_d]^T$ 를 목표경로라 하면, $\tilde{x} = x - x_d$ 는 변수 x 에서의 추적오차가 되며, 다음과 같이 추적오차 벡터로 표현할 수 있다.

$$\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_d = [\tilde{x} \ \dot{\tilde{x}}]^T \quad (4)$$

지금 $s(\mathbf{x}; t)$ 를 다음과 같이 정의하고,

$$s(\mathbf{x}; t) = \left(\frac{d}{dt} + \lambda \right) \tilde{x}, \quad \lambda > 0 \quad (5)$$

(5)에서 $s(\mathbf{x}; t) = 0$ 일때의 시변평면(time-varying surface)을 $S(t)$ 라 정의하면, 모든 $t > 0$ 에 대하여 시스템 궤적이 슬라이딩평면(sliding surface) $S(t)$ 상에 남아 있도록 하는 슬라이딩 조건(sliding condition)은 다음과 같다[7].

$$\frac{1}{2} \frac{d}{dt} s^2 \leq -\eta |s|, \quad \eta > 0 \quad (6)$$

(1)의 시스템에 대한 공칭제어 입력 \hat{u} 는 (5)로부터 $\dot{s} = 0$ 를 만족시키는 제어법칙을 구함으로써 얻을 수 있다. 따라서,

$$\begin{aligned} \dot{s} &= \ddot{\tilde{x}} + \lambda \dot{\tilde{x}} = \ddot{x} - \ddot{x}_d + \lambda \dot{\tilde{x}} = f + bu - \ddot{x}_d + \lambda \dot{\tilde{x}} \\ \therefore \hat{u} &= \hat{b}^{-1} (-\hat{f} + \ddot{x}_d - \lambda \dot{\tilde{x}}) \end{aligned} \quad (7)$$

여기서 \hat{f} 와 \hat{b} 는 각각 f 와 b 의 공칭값이다. (1)에 대하여 슬라이딩 조건 (6)을 만족시키는 제어법칙은 다음과 같이 구할 수 있다[7].

$$u = \hat{u} - k \operatorname{sgn}(s) \quad (8)$$

$$\operatorname{sgn}(s) = \begin{cases} +1 & \text{if } s > 0 \\ -1 & \text{if } s < 0 \end{cases}$$

여기서

$$bk - \eta \geq |f - b \hat{b}^{-1} \hat{f} + (1 - b \hat{b}^{-1})(-\dot{x}_d + \lambda \hat{x})| \quad (9)$$

(8)에서 $k = k(\mathbf{x})$ 를 (9)를 만족하도록 충분히 크게 선택하면, 슬라이딩조건 (6)이 만족한다는 것을 다음과 같이 증명할 수 있다.

$$\frac{1}{2} \frac{d}{dt} s^2 = \dot{s} \cdot s = [(f - b \hat{b}^{-1} \hat{f}) + (1 - b \hat{b}^{-1})(-\dot{x}_d + \lambda \hat{x}) - bk \operatorname{sgn}(s)] \cdot s \leq -\eta |s| \quad (10)$$

(8)의 제어입력은 시스템의 불확실성이나 외란등으로 인하여 슬라이딩평면 $S(t)$ 와 교차되는 제어불연속성(control discontinuity)을 가지고 있다. 이 제어불연속으로 인하여 제어입력에서 발생하는 채터링 현상은 경계층을 도입하여 없앨 수 있다. 경계층범위를 ϕ 라 하면 (8)의 제어법칙은 다음과 같이 표현된다[7].

$$u = \hat{u} - k \operatorname{sat}(s/\phi), \quad \phi > 0 \quad (11)$$

$$\operatorname{sat}(s/\phi) = \begin{cases} s/\phi & \text{if } |s/\phi| \leq 1 \\ \operatorname{sgn}(s/\phi) & \text{otherwise} \end{cases}$$

여기서 $\operatorname{sgn}(\cdot)$ 의 의미는 (8)과 같고, k 는 (9)와 같다.

III. 제어법칙

제어법칙을 유도하기 위하여 먼저 다음의 정리를 고려하여 보자.

보조정리 1: (7)로 부터 다음을 만족하는 상수 $\Psi > 0$ 이 존재한다.

$$b^{-1}(\lambda \hat{x} - \dot{x}_d + f) \leq \zeta \Psi, \quad \forall x, \dot{x} \quad (12)$$

$$\zeta = 1 + |s| + |s|^2$$

증명: [3] 참조.

본 논문에서 제안하는 신경망 슬라이딩모드 제어법칙은 슬라이딩모드이득과 경계층두께를 신경망의 출력으로 얻는 방법이다.

제어법칙은 다음과 같다.

$$u = -k_1 s - \zeta \hat{k} \operatorname{sat}(s/\hat{\phi}), \quad k_1 > 0 \quad (13)$$

$$\hat{k}: \hat{k} \geq \Psi \text{를 만족하도록 하는 신경망 제어기의 출력}$$

$$\hat{\phi}: \hat{\phi} > 0 \text{을 만족하도록 하는 신경망 제어기의 출력}$$

여기서 \hat{k} 는 슬라이딩모드 제어기의 이득, $\hat{\phi}$ 는 경계층 두께로서, \hat{k} 와 $\hat{\phi}$ 는 신경망의 출력이며, ζ 는 (12)와 같고 $\text{sat}(\cdot)$ 의 의미는 (11)과 같다. (13)의 제어법칙을 고려할 때 다음 결과를 얻을 수 있다.

정리 1: (13)의 제어법칙과 (1)의 플랜트로 구성되는 폐루프 시스템에서 s 는 다음의 한계 내에 존재한다.

$$|s| \leq \hat{\phi} \cdot \frac{\zeta \hat{k}}{\hat{\phi} k_1 + \zeta \hat{k}} \leq \hat{\phi} \quad (14)$$

증명: 다음의 Lyapunov 함수를 고려하면,

$$V = \frac{1}{2} s \cdot s \quad (15)$$

$|s| > \hat{\phi}$ 일 경우, s 의 변화는 다음과 같다.

$$\begin{aligned} \dot{V} &= \dot{s} \cdot s \\ &= [f - \ddot{x}_d + \lambda \dot{x} - b k_1 s - \zeta b \hat{k} \text{sgn}(s/\hat{\phi})] s \\ &\leq \zeta b (\Psi - \hat{k}) |s| - b k_1 |s|^2 \end{aligned} \quad (16)$$

(16)에서, $b \geq 0$ 이므로, 신경망 출력 \hat{k} 가 $\hat{k} \geq \Psi$ 를 만족하면 $\dot{V} \leq 0$ 이 되므로, $|s| \leq \hat{\phi}$ 내에 존재하게 됨을 알 수 있다.

$|s| \leq \hat{\phi}$ 일 경우, $b \geq 0$, $\hat{k} \geq \Psi$ 이므로 s 의 변화는 다음과 같다.

$$\begin{aligned} \dot{V} &= \dot{s} \cdot s \\ &= [f - \ddot{x}_d + \lambda \dot{x} - b k_1 s - b \hat{k} s / \hat{\phi}] s \\ &\leq \zeta b \Psi |s| - b k_1 |s|^2 - \zeta b \hat{k} |s|^2 / \hat{\phi} \\ \Rightarrow \dot{V} &\leq (b \zeta \Psi - b \zeta \hat{k} + b \zeta \hat{k}) |s| - b k_1 |s|^2 - \zeta b \hat{k} |s|^2 / \hat{\phi} \\ \Rightarrow \dot{V} &\leq -b (\hat{k} - \Psi) \zeta |s| + b \zeta \hat{k} |s| - b k_1 |s|^2 - \zeta b \hat{k} |s|^2 / \hat{\phi} \\ &\leq -b k_1 |s|^2 + b \zeta \hat{k} |s| (1 - |s| / \hat{\phi}) \end{aligned} \quad (17)$$

(17)을 보면,

$$|s| \geq \hat{\phi} \cdot \frac{\zeta \hat{k}}{\hat{\phi} k_1 + \zeta \hat{k}} \quad (18)$$

일 때는 $\dot{V} \leq 0$ 가 되므로, 이는 즉 $|s|$ 는 (18) 범위 내에 존재하게 됨을 알 수 있다.

△△△

IV. 신경망 슬라이딩모드 제어기의 구조

1. 제어기의 구조

식 (13)에서 신경망제어부의 최종 출력은 슬라이딩모드제어이득 \hat{k} 및 $\hat{\phi}$ 가 됨을 뜻한다. 신경망제어기의 입력은 $|\dot{x}|$ 및 $|\Delta \dot{x}|$ 가 되며, 신경망제어기의 각 층의 가중행렬은 학습에 의하여 off-line으로 구

하게 된다. 신경망 제어부의 출력 \hat{k} 및 $\hat{\phi}$ 는 슬라이딩모드 제어부 출력 u 의 계산에 사용된다. 식 (13) 제어법칙을 포함한 전체 시스템의 구조를 보면 그림 1과 같다.

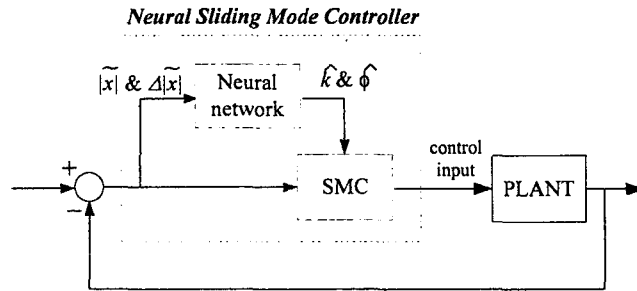


그림 1 Neural SMC 시스템의 구조
Fig. 1 Structure of Neural SMC System

2. 신경망의 구조

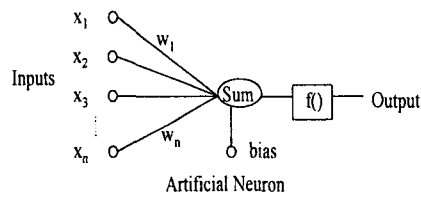


그림 2. 인공 뉴런
Fig. 2. Artificial Neuron

그림 2와 같이 표현되는 인공신경망은 인공뉴런들이 층(layer)을 이루고 있으며 각 층은 입력패턴(input pattern)을 받아들이는 입력층(input layer)과 입력된 패턴에 대한 신경망의 동작 결과를 출력하는 출력층(output layer), 그리고 다층 신경망(multi-layer neural network)의 경우, 입력층과 출력층 사이에 하나 이상의 은닉층(hidden layer)을 가지고 있다.

입력층 신경세포의 입력값들은 가중치와 곱해져 다음 층의 각 신경세포에서 합산되며 합산 값에 활성화 함수가 취해져 해당 세포의 출력이 발생한다. 이를 수식으로 표현하면 다음과 같다.

$$y = f \left[\sum_{k=1}^n W_k x_k + b_k \right] \quad (19)$$

여기서 W 는 가중행렬, b 는 바이어스(bias) 벡터이다.

식 (19)에서 사용하는 활성화함수로는 본 논문에서 사용하는 것은 다음과 같이 계산한다.

a) Linear(Linear activation function) : 출력 = 입력

$$f(x) = x \quad (20)$$

b) Tanh(Hyperbolic tangent function) : 출력의 범위는 -1 과 1 사이이다.

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (21)$$

기타 활성화함수의 종류와 계산방법은 [1]등에 자세히 설명되어 있다.

그림 2의 구조를 적용하면, 슬라이딩모드 제어법칙 (13)의 이득과 경계층 범위를 계산하는 신경망을 구성할 수 있다. 본 논문에서 사용한 신경망의 구성을 나타내면 그림 4와 같다.

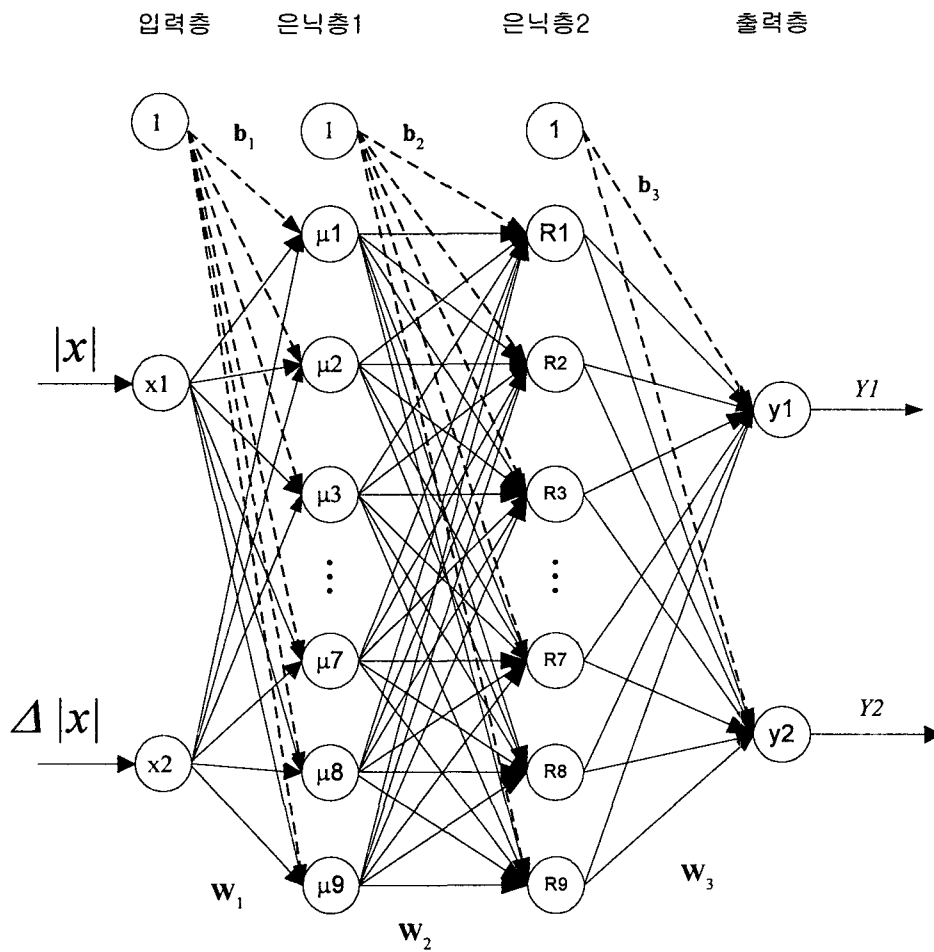


그림 4 가변구조제어를 위한 신경망의 구조

Fig 4 Structure of Network for NVSC

그림 4에서 보면, 출력 계산을 위한 신경망의 구조는, 2개의 입력뉴런과 9개의 hyperbolic tangent neurons으로 구성된 은닉층 1과 9개의 linear neurons으로 구성된 은닉층 2 및 2개의 inear neurons으로 구성된 출력뉴런의 다층 신경망의 형태로 되어 있음을 알 수 있다. 그림 4는 다음과 같이 행렬대수

(matrix algebra)를 사용하여 계산할 수 있다. 이에 따라, 출력은 다음과 같은 matrix 대수식으로 계산한다.

$$y = \text{linear}(W_3 * \text{linear}(W_2 * \tanh(W_1 * x + b_1) + b_2) + b_3) \quad (22)$$

여기서 W_i 는 가중행렬, b_i 는 바이어스를 나타낸다.

3. 신경망의 학습

신경망에서 인공 신경세포의 출력은 연결된 다수의 인공 신경세포들로부터 연결가중치를 통해 전달된 값들의 합산에 의한다. 따라서 신경망 스스로가 외부로부터 주어지는 정보에 대해 어떤 일정한 규칙에 의해 자신의 연결가중치를 조절할 수 있다면, 신경망은 주어지는 정보에 따라 자신의 정보처리 상태를 변경시켜 갈 수 있을 것이다. 이것이 바로 신경망의 학습원리이며, 연결 가중치를 조절하는 일정한 규칙이 바로 신경망의 학습 규칙(learning rule)이다[2,5].

다수의 층으로 이루어진 다층 퍼셉트론을 학습시킬수 있는 방법중 하나는 일반화된 델타규칙으로서, 출력층에서 발생한 에러를 아래층으로 역전파 시켜 가중치를 조절시킬 수 있다. 이러한 이유로, 이 일반화된 델타규칙을 오차 역전파 알고리즘(error backpropagation algorithm)이라고 하며, 이것을 사용하는 신경망을 오차 역전파 모델(error backpropagation model)이라 한다.

신경망의 전반적인 학습과정은 다음과 같다.

- ① 초기값이 잘 설정된 신경망은 쉽게 학습되지만, 그렇지 못한 경우 학습하는데 매우 많은 반복이 필요하거나 아예 학습이 되지않는 수도 있다. 따라서 연결가중치를 -1 에서 $+1$ 사이의 랜덤한 값을 사용하여 초기화 하도록 한다.
- ② 입력패턴을 만들어 신경망을 학습할 입력으로 사용한다.
- ③ 신경망을 동작시킨다.

이때 신경세포의 활성화 함수로 가장 많이 사용되는 시그모이드(sigmoid) 함수나 탄젠트(tangent)함수는 S형태를 가지기 때문에 아무리 큰 입력이 주어지더라도 결코 1이상의 출력을 내지 않으며, 아무리 작은 입력이 주어지더라도 결코 0(또는 -1)이하의 출력을 내지 않는다. 이것은 신경망 내의 어느 한 신경세포가 신경망 전체의 동작을 지배하게 되는 문제를 해결해 준다.

- ④ 출력층 신경세포들의 에러와 델타를 구해 이를 은닉층으로 역전파 한다.
- ⑤ 역전파된 델타로부터 은닉층 신경세포들의 에러와 델타를 구해 이를 역 전파하며 이를 입력층 바로 위층까지 반복한다.
- ⑥ 출력층과 모든 은닉층의 에러와 델타가 구해지면 일반화된 델타규칙에 의해 연결 가중치를 조절한다. 이때 모멘텀(momentum)을 사용하여 신경망의 연결 가중치 조절식에 관성을 줌으로써 학습시간을 단축하고 학습 성능을 향상시킬 수 있다.
- ⑦ 위의 과정을 모든 입력 패턴에 대해 반복한다.
- ⑧ 신경망이 완전히 학습될 때까지 반복한다.

신경망의 출력과 교사신호(Teacher Signal)사이의 MSE(mean square error)값이 정해진 값 이하로 떨어지면 학습이 완료된 것으로 간주하고 지금까지 학습한 연결가중치들을 저장한다.

V. 설계의 예

이 절에서는 [7]에서 예로써 제시된 다음과 같은 시스템을 선택하여, 본 논문에서 제시된 적응제어법칙의 설계와 컴퓨터 시뮬레이션 결과를 보인다.

$$\dot{x} + a(t)x^2 \cos 3x = bu \quad (23)$$

여기서 $a(t)$ 및 $b(t)$ 를 정확하게는 알 수 없지만 다음의 범위 내에서 변화한다고 가정하자.

$$1 \leq a(t) \leq 3, \quad 0.8 \leq b(t) \leq 1.2$$

이때, (3)으로부터 $b_{\min} = 0.8$, $b_{\max} = 1.2$, $\hat{b} = (0.8 \times 1.2)^{1/2} = 0.9798$ 이 되고, (2)의 \hat{f} 및 F 는 다음과 같이 결정된다.

$$\hat{f} = -2.0 x^2 \cos 3x \quad F = 1.0 x^2 |\cos 3x|$$

(23)의 시스템에서 $a(t)$ 및 $b(t)$ 의 실제 값이 $a(t) = |\sin t| + 2$ 및 $b(t) = \cos(5t) * 0.2 + 1.0$ 로 각각 변화하는 시스템을 고려하여 본 논문에서 제시된 적응제어법칙을 적용, 비교하여 본다. 여기서 목표경로는 $x_d = \sin(\pi t/2)$ 으로 한다.

먼저 슬라이딩모드 제어법칙 (8)을 사용하였을 경우 $\lambda = 20$, $\eta = 0.1$ 로 하면 제어법칙은 다음과 같다.

$$\begin{aligned} u &= \hat{u} - k \operatorname{sgn}(s) \\ \hat{u} &= (1/0.9798)(2.0 x^2 \cos 3x + \ddot{x}_d - 20 \dot{x}) \\ k &= (1/0.9798)[(1.5/0.5)^{1/2} * (1.0 x^2 |\cos 3x| + 0.1) + |(1.5/0.5)^{1/2} - 1| * |\hat{u}|] \end{aligned} \quad (24)$$

이때의 추적오차와 제어신호를 그림 5에 보인다. 시뮬레이션 결과를 보면, 추적성능은 우수하지만 제어입력의 채터링이 심하게 발생하는 것을 볼 수 있다.

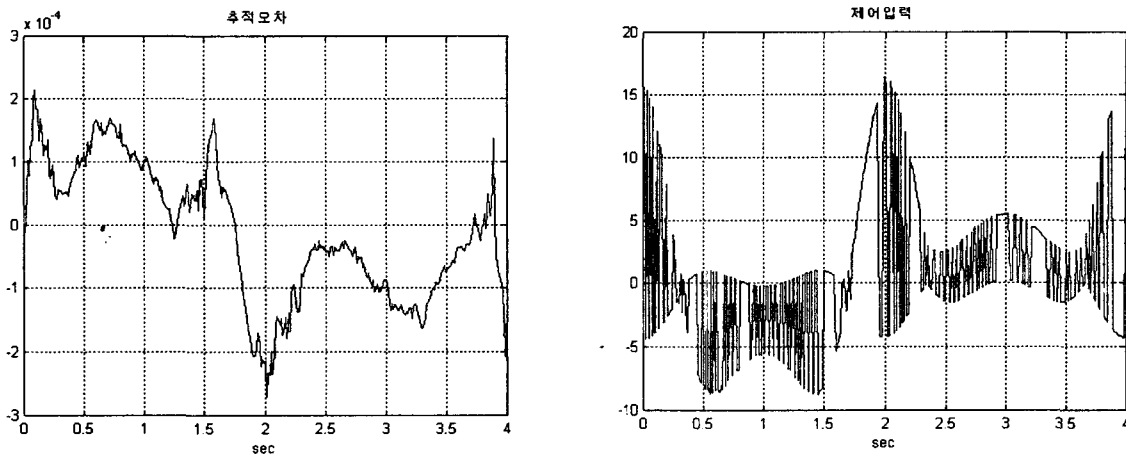


그림 5. 경계층을 적용하지 않은 경우의 추적성능 및 제어입력

Fig. 5. Tracking performance and control input without boundary layer implemented.

다음, 본 논문에서 제시한 신경망제어법칙 (13)과 그림 4의 신경망을 사용하여 시뮬레이션 하였다. 신경망의 학습은 0.001sec의 샘플링시간으로 2초간 역전파오차방법을 사용하여 MATLAB을 이용하여 학습하였다. 학습후의 가중행렬 및 바이어스는 표 1과 같다. 학습후 표 1을 신경망에 적용한 결과, 추적 오차, 제어입력, \hat{k} 및 경계층두께 $\hat{\phi}$ 를 그림 6에 보인다. 시뮬레이션 결과를 보면, 적절한 \hat{k} 및 경계층 두께가 신경망에 의하여 계산되었음을 볼 수 있으며 이에 따라 추적오차는 그림 5 보다 약간 증가하긴 하였으나 여전히 우수한 상태임을 유지하면서도 제어신호에서의 채터링이 사라진 것을 볼 수 있다. 주지할 사항은, 그림 6은 그림 5와는 달리 공칭제어입력을 사용하지 않으므로 복잡한 시스템이나 불확실한 시스템, 또는 MIMO시스템 등에 쉽게 적용이 가능할 것이다.

표 1. 학습후 가중행렬과 바이어스 벡터 결과

Table 1. Weight matrix and bias vector after learning

W1 =		b1 =	b2 =	b3 =				
0.5340	0.3083	0.8107	0.2883	0.0332				
0.2839	0.3933	0.4241	0.4975	0.6574				
0.4484	0.0318	0.0271	0.4139					
0.7523	0.6835	0.5998	-0.9466					
0	0.6877	0	-0.0361					
0.8163	-0.6416	0.4785	0.4259					
0.3594	-0.0899	-0.6443	-0.5284					
0.3008	0.2152	-0.1535	0.2510					
-0.8459	0.9359	0.8511	0.4122					
W2 =								
0.8455	0.5300	-0.7176	-0.7548	-0.2708	0.6474	0.7844	-0.3015	0.2321
0.3109	-0.1950	0.9491	0.5396	0.4375	0.8074	0.7291	0.2894	-0.4555
-0.8823	-0.0777	-0.8074	0.7672	-0.1480	0.8283	0.5242	-0.6272	0.1895
-0.1941	-0.6376	-0.0995	0.7091	0.1593	0.7254	0.3082	-0.7816	0.1015
-0.7443	0.3466	0.6103	-0.2261	0.5097	0.1803	0.1594	-0.6915	0.4634
0.6986	0.1770	-0.6569	-0.3702	0.2393	-0.7647	-0.1683	-0.8048	-0.9595
-0.9429	-0.0262	-0.9834	0.1597	0.3353	0.7168	-0.6707	0.4242	-0.6074
-0.3831	0.1246	-0.0932	0.0050	0.0318	-0.1952	-0.0039	0.0907	0.2312
-0.5924	0.0352	-0.4884	0.7201	0.2529	0.5226	-0.0607	0.2753	-0.4091
W3 =								
0.3263	-0.1346	0.6511	-0.4899	0.3619	-0.3592	-0.7252	-0.2456	-0.6245
-0.2635	0.3215	0.3307	0.2075	0.3822	-0.2292	-0.0563	0.0609	-0.4072

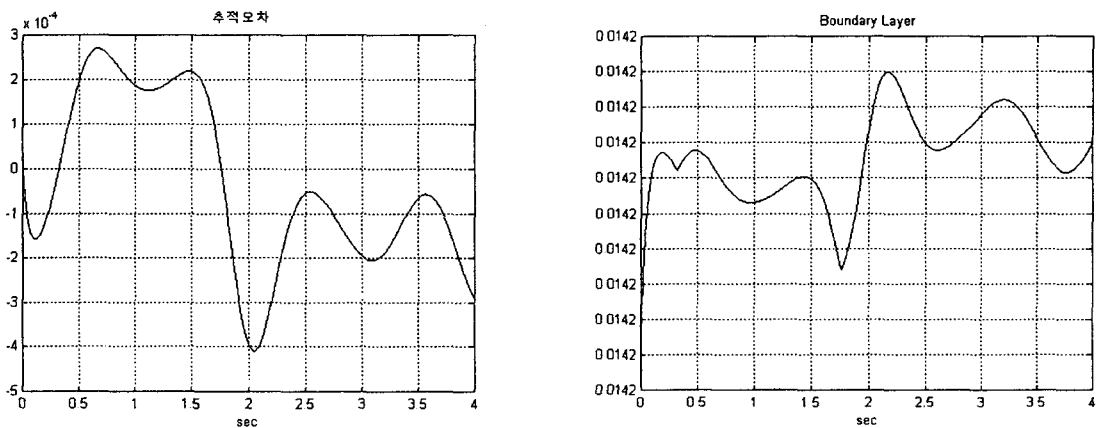
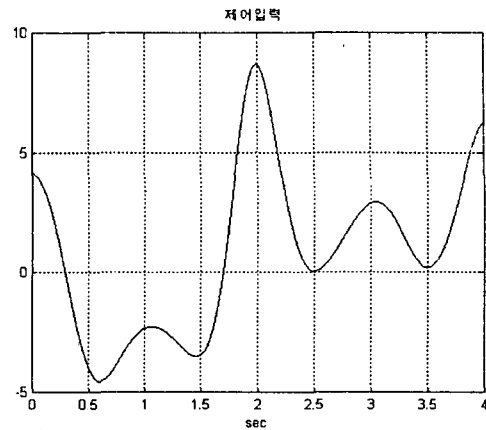
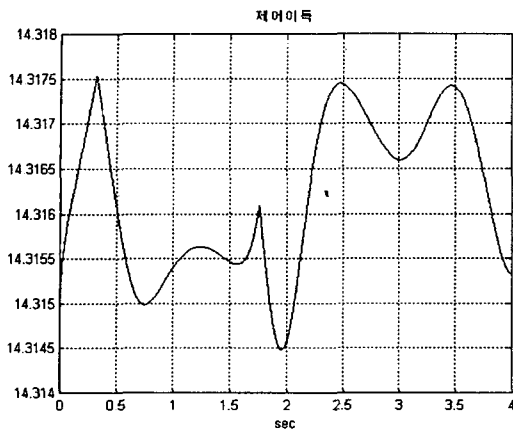


그림 6. 신경망 슬라이딩모드제어의 시뮬레이션 결과

Fig. 6. Simulation results under neural sliding mode control



VI. 결론

본 논문에서는 슬라이딩모드이득 k 와 경계층두께(boundary layer thickness) ϕ 를 신경망을 이용하여 계산하는 신경망 가변구조제어기 설계방법을 제시하였다. 제안된 방법은, 시스템의 불확실성에 대한 슬라이딩모드 이득 k 와 경계층범위 ϕ 를 계산할 수 있는 신경망의 가중행렬을 역전파 오차 학습으로 구함으로서 비선형시스템을 위한 제어기 설계를 용이하도록 하였으며, 이득 k 와 경계층범위 ϕ 를 동시에 계산함으로써 공칭제어입력을 사용하지 않으므로 복잡한 시스템이나 불확실한 시스템, 또는 MIMO시스템 등에 쉽게 적용이 가능하도록 하였다.

참고문헌

- [1] W. Hines, "MATLAB SUPPLEMENT TO Fuzzy and Neural Approach in Engineering," A willy- Interscience JOHN WILEY & SONS INC., 1991
- [2] K. J. Hunt, D. Sbarbaro and R. Zdikowski, PP. J. Gawthrop, "Neural networks for control systems -A survey," *Automatica*, Vol. 28, No. 6, pp.1083-1112, 1992
- [3] 박재삼(J.S.Park), "자기동조 경계층 범위를 갖는 적응 슬라이딩모드 제어(Adaptive Sliding Mode Control with Self-tuning the Boundary Layer Thickness)", 제어.자동화.시스템공학 논문지(*J. of Cont, Auto and Sys. Eng.*), 2000, 게재예정.
- [4] D. Nauck, R. Kruse, "A Neural Fuzzy Controller Learning by Fuzzy Error Backpropagation" *proc. NAFIPS'93*, pp. 388-397, Dec. 14-17, 1992
- [5] D. H. Nguyen, B. Widrow, " Neural networks for self-learning control systems," *IEEE Control Systems Magazine*, Vol. 10, No. 2, pp. 18-23, 1990.
- [6] J.J. Slotine, "Sliding controller design for nonlinear systems", *Int. J. Control.* vol.40, no.2,

pp.421-434, 1984.

- [7] J.J.E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice Hall, 1991.
- [8] V.I. Utkin, "Variable structure systems with sliding modes", *IEEE Trans. on Aut. Cont.* vol AC-22, pp.212-222, 1977.
- [9] V.I. Utkin, *Sliding Modes in Control Optimization*, Springer-Verlag, Berlin, 1992