

비선형 시스템제어를 위한 복합적응 신경회로망

김효규*, 오세영*, 김성권**

* 포항공과대학 전자전기공학과

** 삼성전자 생산기술센터

Composite Adaptive Neural Network Controller for Nonlinear Systems

Hyo-Gyu Kim*, Se-Young Oh*, Sung-Kwun Kim**

* Dept. of Electrical Engineering, Pohang Institute of Science and Technology

** Production Engineering Center, Samsung Electronics

Abstract

In this paper, we proposed an indirect learning and direct adaptive control schemes using neural networks, i.e. composite adaptive neural control, for a class of continuous nonlinear systems. With the indirect learning method, the neural network learns the nonlinear basis of the system inverse dynamics by a modified backpropagation learning rule. The basis spans the local vector space of inverse dynamics with the direct adaptation method when the indirect learning result is within a prescribed error tolerance, as such this method is closely related to the adaptive control methods. Also hash addressing technique, similar to the CMAC functional architecture, is introduced for partitioning network hidden nodes according to the system states, so global neuro control properties can be organized by the local ones. For uniform stability, the sliding mode control is introduced when the neural network has not sufficiently learned the system dynamics. With proper assumptions on the controlled system, global stability and tracking error convergence proof can be given. The performance of the proposed control scheme is demonstrated with the simulation results of a nonlinear system.

1 서론

비선형시스템에 있어서 모델링의 부정확성은 다음의 두종류로 분류할 수 있다. 즉, 구조적인 부정확성(structured or parametric uncertainty)과 비구조적인 부정확성(unstructured uncertainty or unmodeled dynamics)으로 나눌 수 있다. 구조적인 부정확성으로는 정확한 시스템의 모델하에서 부정확한 계수를 가지는 경우이며, 비구조적인 부정확성은 시스템의 설계자가 알지 못하는 효과나 모델하기 너무 복잡한 경우가 그것이다. 고전적인 적응제어(adaptive control)기법은 시스템이 동작점 근방에서는 선형이라는 가정과 비선형성을 미리 알 수 있다는 가정하에 기본적으로 순환(recursive) 적용의 선형제어기를 사용하였다. 그러나 이러한 강제적인 시스템의 선형화의 가정은 안정도(stability)나 정확성(accuracy), 강인성(robustness)면에서 많은 문제점을 야기하였다. 따라서 이러한 문제를 해결하기 위해서는 시스템의 비선형성을 제어기에서는 적절히 고려하여야 한다. 최근의 강인성제

어(robust control), 적응제어 그리고 신경제어(neuro control) 기법들의 발전은 이러한 면에서 많은 유용한 결과들을 보여주었다 [1, 9, 11]. 강인성제어는 시스템 계수의 한계범위에서 선형시스템 유형으로 안정화시키는 일정이득 제어기의 설계기법이며, 적응제어는 계수를 모르는 시스템을 안정화하기 위해 제어기의 특성을 조정하는 기법이다. 그리고 신경제어는 입력력 데이터쌍으로부터 시스템의 동력학을 학습하고, 학습한 정보를 가중치(weights)로써 분산 저장하며 이를 제어에 이용한다. 신경회로를 이용한 비선형시스템의 제어는 학습기능과 임의의 비선형함수 근사(approximation) 특성[2]으로 인해 최근 많은 연구가 있었다. 그러나 순수한 신경제어 방법은 몇가지 결점이 있는데 그것은 제어시스템의 안정성 증명 이 어렵고, 강인성과 수렴특성이 불명확하다는 것이다. 또한 일반화특성도 신경제어의 하나의 과제이다.

본 연구에서는 이러한 문제들을 해결할 수 있는 방법을 제시하기 위해 Sanner[10]가 제안한 것과 같이 슬라이딩모드 제어(sliding mode control) 기법과 적응제어 기법을 신경제어에 결합하는 하이브리드(hybrid) 신경제어 기법을 제안하였다. 또한 신경회로의 가중치 조정 방법으로는 복합적응기법 (composite adaptive method)을 제안하였다. 복합적응기법은 예측오차에 의한 학습 (Prediction Error Based Learning : PEBL)과 추적오차에 의한 학습 (Tracking Error Based Learning : TEBL)으로 분류할 수 있는데, 이는 시스템의 파라메타 정보가 추적오차 뿐만 아니라 예측오차에도 포함되어 있기 때문이다 [12]. 슬라이딩모드 제어 기법은 신경회로가 시스템의 동력학을 충분히 학습하지 못한 동안에 제어 시스템의 균등한 안정도(uniform stability)를 위해 도입되었다. 또한 전체 제어시스템을 국부공간(local space)으로 해석하고 이 결과를 전공간(global space)으로 확장하기 위해 CMAC의 함수구조 기법을 도입하여 입력상태 변수의 값에 따라 신경회로의 중간층노드(hidden nodes)를 선택한다. 이를 기본으로 PEBL에 의해 시스템의 동력학의 국부공간에서의 비선형 basis를 찾고, PEBL에 의해 동력학의 추정이 지정된 오차 한계치 이내에 들면 TEBL에 의해 비선형 basis는 동력학의 국부공간을 span하게 된다. 따라서 PEBL은 신경회로망 기법의 일종이며, TEBL은 적응제어 기법에 해당된다.

2 신경회로의 함수 근사이론

어떤 함수 $f(x)$ 를 근사화하는 하나의 중간층을 가지는 신경회로의 수식 표현을 다음과 같이 정의 한다.

$$\hat{f}(x, P) = W^2\Phi(x, W^1)$$

$$= \sum_{i=1}^{d_1} w_i^2 \phi_i(x, W^1) + w_0^2 \quad (1)$$

$$\phi(x, W^1) = \phi\left(\sum_{i=1}^{d_1} w_i^1 x_i + w_0^1\right) \quad (2)$$

여기서 $P = [W^1 W^2]$, $W^i = [w_0^i w_1^i w_2^i \dots w_{d_i}^i]^T$, for $i = 1, 2$ 이며, W^1 은 신경회로의 중간층 가중치 행렬, W^2 는 출력층 가중치 행렬이며, w_0^i 는 바이어스(bias) 가중치 벡터를 나타낸다. $\phi(\cdot) : R \rightarrow (-1, 1)$ 은 단조증가 함수로서 $\tanh(\cdot)$ 과 같은 시그모이드(sigmoid) 함수를 사용한다.

Hornik *et al.* [2] 은 어떤 compact 영역내의 임의의 연속함수는 하나의 중간층을 가지는 신경회로에 의해 임의의 오차 이내로 근사화될 수 있다는 것을 증명하였다. 즉, 신경회로는 충분히 많은 개수의 중간층 뉴런에 의해 주어진 임의의 $\epsilon (> 0)$ 에 대해 다음과 같이 정의될 수 있다.

$$\|\hat{f}(x, P) - f(x)\| < \epsilon \quad \text{for all } x \in D \quad (3)$$

$\hat{f}(x, P)$ 는 $f(x)$ 의 신경회로에 의한 근사화 함수이며, D 는 유한 차수 norm 이 정의된 벡터공간의 compact 영역을 표시한다. 신경회로의 가중치를 조정하기 위해 다음과 같이 오차 목적함수 E 를 정의하고

$$E = \frac{1}{2} \|\hat{f}(x, P) - f(x)\|^2 \quad (4)$$

E 를 극소화하기 위해 각 학습데이터의 입력에 따라 신경회로의 가중치를 다음과 같이 경사조정법(gradient decent method)에 의해 조정하게 된다.

$$P(k+1) = P(k) - \eta \frac{\partial E}{\partial P} \Big|_{P=P(k)} \quad (5)$$

여기서 η 는 학습률(learning rate) 을 나타낸다. 만약 η 가 어떤 조건을 만족하게 되면 (일례로 충분히 작거나) 다음의 관계가 성립하게 된다 [1, 7].

$$\|\hat{f}(x_k, P(k+1)) - f(x_k)\| \leq \|\hat{f}(x_k, P(k)) - f(x_k)\| \quad (6)$$

이와같은 학습방법에 의해 다음과 같이 정의된 평균오차(average error) 는 N 이 충분히 커짐에 따라 감소하면서 수렴할 수 있다고 알려져 있다.

$$\frac{1}{2N} \sum_{k=1}^N \|\hat{f}(x_k, P(k)) - f(x_k)\|^2$$

$$x_k \in D \quad \text{for } k = 1, \dots, N$$

따라서 이러한 몇가지 조건하에서 신경회로는 임의의 연속함수를 지정된 임의의 오차 이내로 근사화할 수 있게 된다.

Backpropagation 기법으로 알려진 generalized delta rule 은 어떤 compact 영역내의 임의의 연속함수를 신경회로로 근사화하기 위한 하나의 효과적인 방법으로서 잘 알려져 있다.

3 시스템 동력학의 구역화 이론

제어대상 시스템 동력학의 전영역은 보통 매우 복잡해서 신경회로로 전체동력학 영역을 근사화하기 위해서는 많은 수의 중간층 뉴런이 필요하게 되며, 학습시간이 길어지고, 실시간제어가 어려워지며, 새로운 제어를 학습함에 따라 과거의 정보를 잊어버리는 경우가 생기게 된다. 이러한 문제를 해결하는 방법으로 그림 1과 같이 중간층의 뉴런을 시스템의 상태변수 크기에 따라 효과적으로 구역화(partitioning) 하는 기법을 제안한다. 이러한 구역화에 의해 작

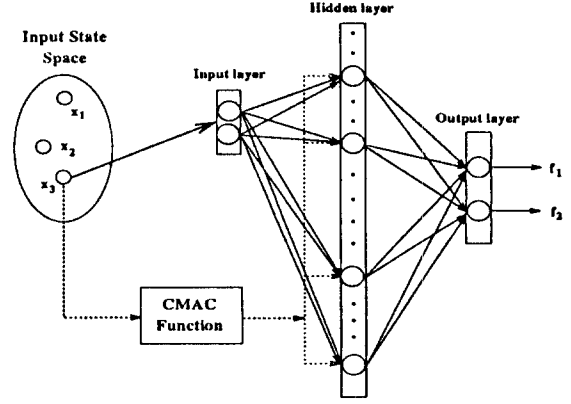


그림 1: CMAC 함수구조를 이용한 신경회로의 구역화

은수의 뉴런만을 사용함으로써 학습시간을 단축하고, 실시간제어가 가능하며, 새로운 제에는 다른 구역의 뉴런을 학습에 사용함으로써 정보손실을 억제할 수 있다. 그리고 인접구역을 서로 효과적으로 중첩함으로써 중간층 뉴런수를 많이 줄일 수 있다.

본 연구에서는 이러한 구역화의 기법으로 CMAC(Cerebellar Model Arithmetic Computer)의 함수구조[4]에서 사용된 기법을 사용하였다. 이 기법은 미리 설정된 상태변수의 최대, 최소값을 기준으로 입력상태 변수를 양자화(quantization)하여 segment mapping을 하고, 이 segment에 의해 논리적 주소(logical address)를 만들고, 이를 hash mapping에 의해 물리적 주소(physical address)를 만들어 중간층 뉴런을 선택하게 된다. 구역화된 중간층 뉴런은 시스템 동력학의 국부 compact 영역을 표현하며 이 영역안에서의 동력학 함수를 근사화하게 된다.

4 Basis 탐색이론

신경회로의 함수근사 이론을 바탕으로 하여 실제적인 함수 근사관계를 다음과 같이 표현할 수 있다.

$$\begin{aligned} f(x) &= \hat{f}(x, P^*) + \epsilon_f \\ &= \hat{f}(x, \hat{P}) + \epsilon_x + \epsilon_f \\ &= \hat{f}(x, P_0) + \epsilon_s + \epsilon_x + \epsilon_f \\ &< \hat{f}(x, P_0) + \epsilon \end{aligned} \quad (7)$$

여기서 $P^* = [W^{*1} W^{*2}]$ 는 정확한 목적가중치(target weight), $\hat{P} = [W^1 W^2]$ 는 최적가중치(optimal weight), $P_0 = [W_0^1 W_0^2]$ 는 offline 학습에 의해 미리 학습한 nominal 가중치를 나타낸다. 그리고 ϵ_f 는 유한개수의 중간층 뉴런으로 인한 한계근사오차를, ϵ_x 는 외란이나 잡음으로 인한 가중치의 drift를 방지하기 위해 사전에 설정한 dead zone을, ϵ_s 는 중간층 출력률 비선형 basis로 하고 출력가중치를 nominal 값으로 하여 P_0 에서 \hat{P} 로 가는 출력가중치의 변위, \hat{W}^2 를 나타낸다. 즉, $f(x)$ 가 $B_\epsilon = \{x : \|x - x_0\| < \epsilon\}$ 로 표현되는 hash addressing에 의해 선택된 국소 compact 영역내에서 충분히 smooth하며, 이 영역내에서는 offline 학습이 된 d_2 개의 신경회로의 중간층 출력률 basis로 하여 ϵ_s 는 다음과 같이 표현될 수 있다고 가정한다.

$$\begin{aligned} \epsilon_s &= \sum_{i=1}^{d_2} \hat{w}_i^2 \phi_i(x, W_0^1) + w_0^2 \\ &= \hat{W}^2 \Phi(x, W_0^1) \end{aligned} \quad (8)$$

따라서 식(7)을 다시 쓰면,

$$\begin{aligned} f(x) &= W_0^2 \Phi(x, W_0^1) + \bar{W}^2 \Phi(x, W_0^1) + \epsilon_z + \epsilon_f \\ &= (W_0^2 + \bar{W}^2) \Phi(x, W_0^1) + \epsilon_z + \epsilon_f \end{aligned} \quad (9)$$

따라서 최적의 근사함수는 다음과 같이 된다.

$$\begin{aligned} \hat{f}(x, \hat{P}) &= (W_0^2 + \bar{W}^2) \Phi(x, W_0^1) \\ &= \hat{f}(x, P_0) + \hat{f}(x, \hat{P}) \end{aligned} \quad (10)$$

본 연구에서 제안하는 복합적용 학습방식에서는 예측오차에 의한 학습(PEBL)에 의해 $\hat{f}(x, P_0)$ 를 backpropagation 방법의 해 먼저 학습한다. 그러면 PEBL 후의 나머지 오차는 식(10)으로부터 다음과 같이 표현된다.

$$\hat{f}(x, \hat{P}) - \hat{f}(x, P_0) = \bar{W}^2 \Phi(x, W_0^1)$$

추적오차에 의해 학습(TEBL)은 PEBL의 결과가 지정된 오차범위 이내로 예측오차가 수렴하게 되면, 이 오차범위 내에서는 시스템이 충분히 파라메타 선형화(parameter linearization)가 되었다고 가정하고 기존의 적응제어 방식에 의해 $\hat{f}(x, \hat{P})$ 의 출력가중치 \bar{W}^2 만이 조정된다.

5 복합적용 제어구조 및 설계

5.1 제어구조

시스템동역학의 지정된 범위의 상태변수 구역내에서 안정한 학습기능을 확보하고 제어하기 위해 Sanner[10]가 제안한 방법과 같이 신경회로에 의한 적응제어기법과 슬라이딩모드 제어기법을 변조함수에 의해 적절히 선택하는 그림.2와 같은 하이브리드 제어기법을 도입한다.

여기서 제어법칙은 다음과 같다.

$$u(t) = u_{pd}(t) + m(t)u_{nn}(t) + (1 - m(t))u_{sl}(t) \quad (11)$$

여기서 $u_{pd}(t)$ 는 선형상태변수 제어입력, $u_{nn}(t)$ 는 신경회로에 의한 적응제어 입력, $u_{sl}(t)$ 는 슬라이딩모드 제어입력이다. 그리고 $m(t)$ 는 하이브리드 제어를 위한 변조함수로서 PEBL에 의한 신경회로의 시스템 역동력학의 근사오차 e_p 에 의해 다음과 같이 결정된다.

$$\begin{aligned} m(t) &= 0 & \text{when } e_p > \epsilon_p \\ m(t) &= 1 & \text{when } e_p < \epsilon_p - \varphi \end{aligned}$$

로 결정하며, 제어방법의 변조시의 제어입력 불연속성을 방지하기 위해 경계층 φ 를 도입하고 경계층 두께의 근사오차 범위에서는 변조함수를 다음과 같이 결정한다.

$$0 < m(t) < 1 \quad \text{when } \epsilon_p - \varphi < e_p < \epsilon_p$$

그리고 유한개의 뉴론에 의한 한계근사오차와 잡음과 외란에 의한 가중치의 drift를 방지하기 위해 $e_p < \epsilon_f + \epsilon_z$ 일 때는 PEBL 동작을 멈추게 되며, 추적오차 e 가 추적한계(tracking dead zone) 이하일 때는 TEBL의 동작을 멈추게 되어 신경제어기는 전방향(feedforward) 동작만을 하게 된다.

5.2 복합적용 제어기 설계

여기서는 다음과 같은 표준형(canonical form)의 비선형 시스템을 대상으로 하여 복합적용(composite adaptive) 제어기 설계문제를 다룬다.

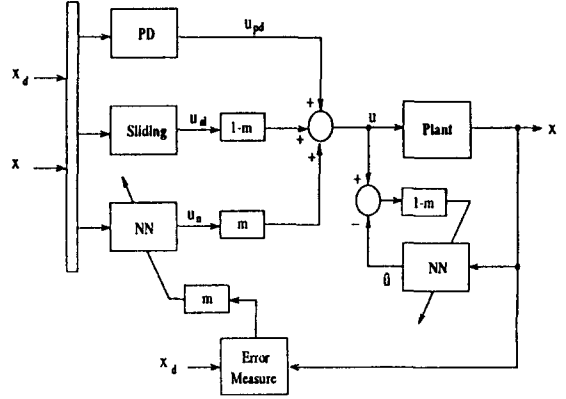


그림 2: 복합적용 제어구조

$$\dot{x}^{(n)}(t) = \alpha(x(t)) + \beta(x(t))u(t) \quad (12)$$

여기서 $x(t) = [x(t) \ \dot{x}(t) \ \dots \ x^{(n-1)}(t)]$ 이며, $u(t)$ 는 시스템의 제어입력, $\alpha(x(t))$ 와 $\beta(x(t))$ 는 모르는 비선형 함수이다.

그리고 이 시스템의 역동력학(inverse dynamics)을 다음으로 나타낸다.

$$u(t) = f(x(t)) + g(x(t))x^{(n)}(t) \quad (13)$$

여기서 $f = -\beta^{-1}\alpha$, $g = \beta^{-1}$ 이다. 식(13)의 신경회로에 의해 근사화된 역동력학은 다음으로 표시한다.

$$\hat{u}(t) = \hat{f}(x(t), P) + \hat{g}(x(t), Q)x^{(n)}(t) \quad (14)$$

여기서 $P = [W^1 \ W^2]$ 는 $f(x)$ 를, $Q = [V^1 \ V^2]$ 는 $g(x)$ 를 근사하는 신경회로의 중간층과 출력층의 가중치 행렬이다.

5.2.1 PEBL

신경회로에 의한 예측오차 e_p 는 다음과 같다.

$$\begin{aligned} e_p &= \hat{u} - u \\ &= \hat{f}(x, P) - f(x) + (\hat{g}(x, Q) - g(x))x^{(n)}(t) \end{aligned} \quad (15)$$

오차 목적함수 E 를 다음과 같이 정의하면,

$$E = \frac{1}{2} e_p^T e_p$$

P, Q 의 각각의 가중치에 대한 E 의 경사(gradient)는

$$\begin{aligned} \frac{\partial E}{\partial P} &= \left(\frac{\partial \hat{f}}{\partial P} \right)^T e_p \\ \frac{\partial E}{\partial Q} &= \left(\frac{\partial \hat{g}}{\partial Q} x^{(n)} \right)^T e_p \end{aligned} \quad (16)$$

이는 기존의 backpropagation 방법을 약간 수정하여 구할 수 있다. 이에 의해 다음과 같은 경사조정법으로 각각의 가중치를 조정한다.

$$P(k+1) = P(k) - \eta_p \left. \frac{\partial E}{\partial P} \right|_{P=P(k)}$$

$$Q(k+1) = Q(k) - \eta_g \frac{\partial E}{\partial Q} \Big|_{Q=Q(k)} \quad (17)$$

식 (15)에서 $x^{(n)}(t)$ 가 충분히 다양하게 움직이면(persistent excitation) e_p 가 극소화될 때 각각의 항이 극소화될 수 있다. 이렇게 하여 PEBL에서는 4장에서 증명한 것처럼 국소구역 각각에서의 $\alpha(x)$ 와 $\beta(x)$ 의 basis 및 nominal 가중치를 학습하게 된다.

5.2.2 TEBL

여기서는 PEBL에 의해 구해진 비선형 basis를 가지고 선형 파라메타화된 제어시스템을 안정화시키고 추적오차가 수렴하는 TEBL 제어를 설계한다. 식(12)에서와 같이 입출력 관계를 가진 동력학 시스템에서 비선형성을 제거하고 선형 입출력 관계, 즉, $x^{(n)} = v$ 를 가지도록 $u = g^{-1}(v - f)$ 의 제어입력을 가하는 방법이 궤환선형화(feedback linearization) 방법이다. 여기서 원하는 궤적 x_d 를 추적하기 위해 v 는 다음의 제어법칙을 따른다.

$$\begin{aligned} v &= x_d^{(n)} - k_1 e - k_2 \dot{e} - \dots - k_{n-1} e^{(n-1)} \\ &= x_d^{(n)} - k^T e \end{aligned} \quad (18)$$

여기서 $e = x - x_d$ 로서 상태변수 추적오차를 나타내며, $k^T = [k_1 \ k_2 \ \dots \ k_{n-1}]$ 는 p 를 Laplace 변수라고 할때, $\Gamma(p) = p^{(n)} + k_{n-1}p^{(n-1)} + \dots + k_1$ 가 Hurwitz 다항식이 되도록 선택하면 추적오차는 지수함수적으로 수렴하게 된다. 그러나 현실적으로 시스템에서 비선형성을 미리 정확하게 알 수 없으므로 제어입력을 $u = \hat{g}(x_d^{(n)} - k^T e) + \hat{f}$ 로 사용하는 경우 시스템의 오차동력학 식은 다음과 같이 된다.

$$e^{(n)} = -k^T e + (I - \hat{g}^{-1}g)x^{(n)} + \hat{g}^{-1}(\hat{f} - f) \quad (19)$$

이를 상태방정식으로 표현하면 다음과 같이 쓸 수 있다.

$$\dot{e}(t) = K e(t) + b \gamma(x^{(n)}(t), x(t)) \quad (20)$$

여기서

$$\begin{aligned} e &= [e \ \dot{e} \ \dots \ e^{(n-1)}]^T \\ K &= \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -k_1 & -k_2 & -k_3 & \dots & -k_{n-1} \end{bmatrix} \\ b &= [0 \ 0 \ 0 \ \dots \ 1]^T \end{aligned} \quad (21)$$

$$\gamma(x^{(n)}, x) = (I - \hat{g}^{-1}g)x^{(n)} + \hat{g}^{-1}(\hat{f} - f) \quad (22)$$

TEBL과 슬라이딩모드 제어를 위한 오차측정(error metric)을 다음과 같이 정의한다 [11].

$$s(t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} e(t) \quad (23)$$

즉,

$$\begin{aligned} s(t) &= e^{(n-1)} + (n-1)\lambda e^{(n-2)} + \dots + \lambda^{n-1} e \\ &= \lambda^T e(t) \end{aligned} \quad (24)$$

$s(t) = 0$ 는 R^n 에서의 추적오차 벡터가 지수함수적으로 수렴하는 hyperplane을 나타낸다.

이상의 수식들로부터 다음의 결론을 내릴 수 있다. 만약 $e(0) = 0$ 라고 가정하면 다음이 성립한다. ($e(0) \neq 0$ 일 경우는 초기조건 의 효과를 독립적으로 추가하면 됨),

$$\begin{aligned} \forall t \geq 0, \quad |s(t)| &\leq |\gamma| \\ \Rightarrow \forall t \geq 0, \quad |e^{(i)}| &\leq 2^i \lambda^{i-n-1} |\gamma|, \quad i = 0, \dots, n-1 \end{aligned}$$

PEBL에서는 $|\gamma|$ 를 극소화하는 가중치를, TEBL에서는 $|s(t)|$ 를 극소화하는 가중치를 학습하게 되어 결국 복합적용 제어방식에 의해 시스템 역동력학 함수의 근사화 뿐만 아니라 추적오차도 극소화할 수 있다는 것을 알 수 있다.

5.2.3 적응제어기 설계

TEBL 적응제어기의 설계는 Sanner 에서 제안한 방법을 도입하여 정의된 상태공간 전체영역에서 안정적인 제어법칙과 학습법칙을 찾는다. 식 (24)의 오차측정을 다시 쓰면,

$$s = x^{(n-1)} - x_r^{(n-1)} \quad (25)$$

여기서 $x_r^{(n-1)}$ 은 다음으로 정의된다.

$$x_r^{(n-1)} = x_d^{(n-1)} - (n-1)\lambda e^{(n-2)} - \dots - \lambda^{n-1} e$$

이를 한번 더 미분한 $x_r^{(n)}$ 은 다음과 같다.

$$x_r^{(n)} = x_d^{(n)} - (n-1)\lambda e^{(n-1)} - \dots - \lambda^{n-1} \dot{e} \quad (26)$$

그리고 슬라이딩 모드 제어시 chattering 방지를 위한 dead zone 의 도입을 위해 다음과 같이 s_Δ 를 정의한다.

$$s_\Delta(t) = s(t) - \psi \text{ sat} \left(\frac{s(t)}{\psi} \right) \quad (27)$$

그러면 오차측정의 시간에 대한 도함수는 다음과 같다.

$$g \dot{s} = \hat{g} x_r^{(n)} - \hat{f} + u + (g - \hat{g}) x_r^{(n)} - (f - \hat{f}) \quad (28)$$

TEBL의 적응제어를 위해 다음의 제어법칙을 사용한다.

$$u = -k_D s - \frac{1}{2} M_2 \|x\| s_\Delta + m(\hat{g} x_r^{(n)} + \hat{f}) + (m-1) u_{sl} \quad (29)$$

여기서 k_D 는 g 와 같은 부호를 가지는 상수 이득이며, u_{sl} 은 슬라이딩모드 제어입력으로 다음과 같다.

$$u_{sl} = -k_{sl} \text{ sat} \left(\frac{s(t)}{\psi} \right) \quad (30)$$

그리고 $M_2 \|x\|$ 는 다음과 같이 g 의 미분에 대한 상한으로 정의된다.

$$|\dot{g}(x)| \leq M_2 \|x\| \quad (31)$$

이 제어법칙을 사용하면 추적오차 동력학은 다음과 같이 표현된다.

$$\begin{aligned} g \dot{s} &= -k_D s - \frac{1}{2} M_2 \|x\| s_\Delta + m[(\hat{g} - g)x_r^{(n)} + \hat{f} - f] \\ &\quad + (1-m)(u_{sl} + g x_r^{(n)} - f) \end{aligned} \quad (32)$$

출력가중치의 학습법칙을 다음과 같이 선택하고,

$$\begin{aligned} \dot{v}_i^2 &= -m \eta_g \text{sgn}(g) s_\Delta \Phi_i(x, V_0^1) x_r^{(n)} \\ \dot{w}_i^2 &= -m \eta_f \text{sgn}(g) s_\Delta \Phi_i(x, W_0^1) \end{aligned} \quad (33)$$

그리고 다음의 Lyapunov 함수를 정의하면,

$$V = |g| s_\Delta^2 + \frac{1}{\eta_g} \sum_{i=1}^{d_{v2}} (v_i^2)^2 + \frac{1}{\eta_f} \sum_{i=1}^{d_{w2}} (w_i^2)^2 \quad (34)$$

이의 시간에 대한 도함수가 음의 조건을 만족하게 되어 추적오차가 수렴하게 된다 [10].

6 시물레이션

본 연구의 이론을 규명하기 위해 Sanner[10] 에서 사용한 것과 같은 표준형의 시스템을 대상으로 시물레이션 하였으며, 똑같은 제어를 추종하는 문제를 다루었다.

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= \alpha(x) + \beta(x)u\end{aligned}\quad (34)$$

여기서

$$\begin{aligned}\alpha(x) &= 4 \frac{\sin(4\pi x)}{\pi x} \left(\frac{\sin(\pi - \dot{x})}{\pi \dot{x}} \right)^2 \\ \beta(x) &= 2 + \sin(3\pi(x - 0.5))\end{aligned}\quad (35)$$

시스템의 원하는 위치 추적레저 x_d 는 0.4Hz의 주파수에 시간평균이 0.5 인 구형과 입력을 $(1 + s/10)^{-3}$ 의 전달함수를 갖는 저역역파기(low-pass filter)에 가했을 때의 출력을 사용하였다.

여기서 상태변수의 상태변수의 운동범위를 $-0.5 \leq x \leq 2.0$, $-4.0 \leq \dot{x} \leq 4.0$ 으로 설정하고 각각 8 비트 양자화(quantization)하여 CMAC 함수구조의 입력으로 하였다. 그리고 시스템의 역동력학 $f(x, \dot{x})$ 와 $g(x)$ 를 근사화하는 신경회로의 입력층 뉴런의 개수는 각각 2개, 1개이며, 중간층의 구역화 뉴런은 각각 10개, 7개씩 선택되도록 하였으므로 중간층 뉴런의 전체개수는 각각 7290개, 266개씩 사용하였다. 그리고 출력층 뉴런의 개수는 각각 1개이다.

제어변수의 설정치로는 $k_D = \lambda = 10$, $k_{s1} = 20$, $\psi = 0.5$, 그리고 $M_2 = 10$ 으로 두었으며, $\eta_p = 0.005$, $\eta_g = 0.0001$, 그리고 $\eta_f = 0.01$, $\eta_g = 0.001$ 로 설정하였다. 변조함수 $m(t)$ 의 선택을 위한 예측오차의 크기판별은 실제 제어입력에 대한 예측오차의 비로써 하였다. 즉, $\epsilon_p = 0.2$ 로써 20%의 비율로, 경계층인 $\varphi = 0.1$, 예측오차 한계(prediction error dead zone)는 0.01로써 설정하였으며, 추적오차 한계는 $\|x\| = 0.0005$ 로 설정하였다.

그림.3과 그림.4는 각각 x, \dot{x} 오차의 학습횟수에 대한 rms 값을 보여주며, 그림.5는 예측오차에 대한 값을 보여준다. 그리고 그림.6은 최종 학습시 위치오차의 마지막 4회 구간을 나타내주고 있다.

7 결론 및 향후연구 과제

본 연구에서는 신경회로에 의한 비선형시스템의 안정한 제어기법을 제시하였다. 즉, 슬라이딩모드 제어기법을 결합한 복합제어 방식과 CMAC 함수구조를 이용한 구역화 기법을 제시하여 시스템의 동력학을 학습함으로써 넓은 상태변수 구간에서도 안정하며 실시간 제어가 가능한 기법을 제시하였다. 제안된 제어기법의 시물레이션 결과에서 보듯이 추적오차와 예측오차의 rms 값이 안정적으로 수렴한다는 것을 알 수 있다. 본 연구의 결과는 많은 수정없이 로보트제어 등에도 이용될 수 있다. 그리고 최종 추적오차의 결과는 Sanner[10]에서의 결과와 비교할 만하다. 또한 사전 offline 학습에 의해 학습시간을 많이 줄일 수 있다.

본 연구의 남은 과제로는 본문에서 제시하는 몇가지의 가정을 좀더 합리화시켜야 한다는 것과 상태공간의 구역화에 필요한 전체 뉴런의 개수를 효과적으로 설정할 수 있는 제안이 필요하다. 또한 학습의 정확한 수렴을 위한 다양한 학습 임출력상의 조건 즉, persistent excitation 조건을 만족시키는 학습연구가 요구된다.

참고 문헌

- [1] Chen, F. C., "Back-propagation neural networks for nonlinear self-tuning adaptive control," *IEEE Control System Magazine*, pp. 44-48, Apr. 1990.
- [2] Hornik, K., M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359-366, 1989.
- [3] Hancock, E. T., F. Fallside, "Stable control of nonlinear systems using neural networks," *Int. J. of Robust and Nonlinear Control*, vol. 2, pp. 63-86, 1992.
- [4] Handelman, D. A., S. H. Lane, J. J. Gelfand, "Characteristics of integrated rule-based systems and neural networks enabling robotic skill acquisition," Proc., of the BIRA Seminar on *Neural Networks*, Ghent, Belgium, Apr. 1989.
- [5] Kim, H. G., S. Y. Oh, "Decomposed self-tuning manipulator controller based on neural networks," *Proceeding of KERC Workshop*, Daejeon, Korea, pp 146-153, Jan. 1993.
- [6] Levin, A. U, K. S. Narendra, "Control of nonlinear dynamical systems using neural networks: controllability and stabilization," *IEEE Tran. on Neural Networks*, vol. 4, no. 2, pp. 192-206, Mar. 1993.
- [7] Ljung, L., T. Soderstrom, *Theory and Practice of Recursive Identification*, MIT Press, 1983.
- [8] Narendra, K. S., A. M. Annaswamy, *Stable Adaptive Systems*, Prentice-Hall International, 1989.
- [9] Psaltis, D., A. Sideris, and A. A. Yamamura, "A multilayered neural network controller," *IEEE Control System Magazine*, pp. 17-21, Apr. 1988.
- [10] Sanner, R. M., and J.-J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. on Neural Networks*, vol. 3, no. 6, pp. 837-863, Nov. 1992.
- [11] Slotine, J.-J. E., and W. Li, *Applied Nonlinear Control*, Prentice Hall International, 1991.
- [12] Slotine, J.-J. E., and W. Li, "Composite adaptive control of robot manipulators," *Automatica*, 25, 4, 1989.

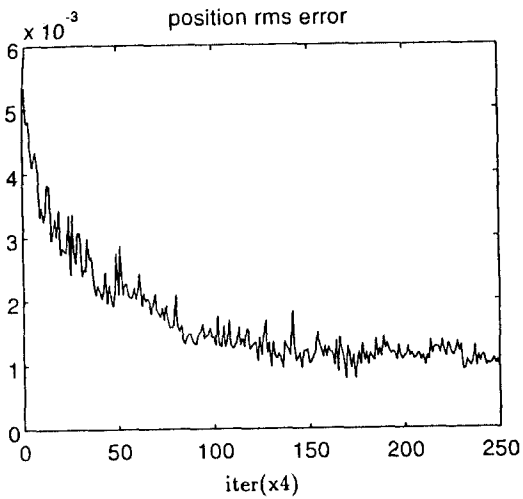


그림 3: 학습횟수에 대한 위치추적 rms 오차

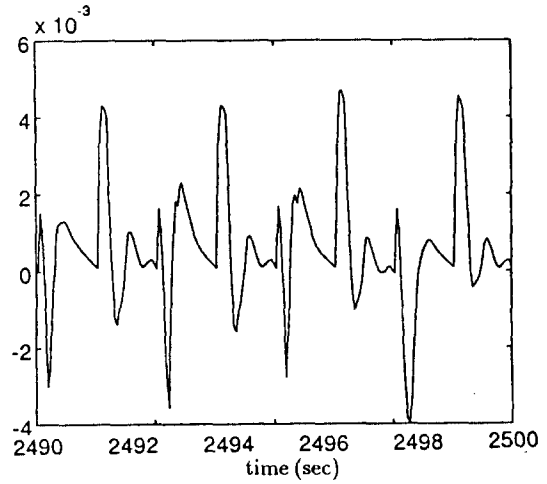


그림 6: 1000회 학습후의 위치 오차

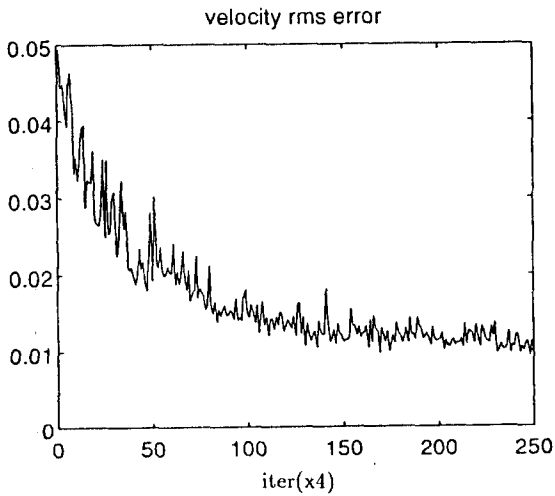


그림 4: 학습횟수에 대한 속도추적 rms 오차

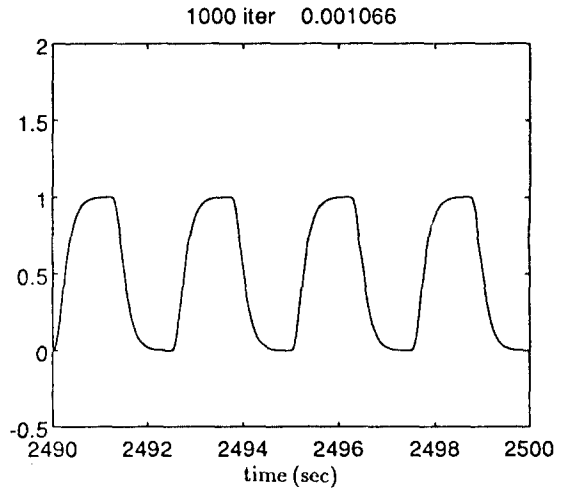


그림 7: 1000회 학습후의 원하는 위치궤적과 추적궤적
(두 궤적이 완전히 겹쳐있음)

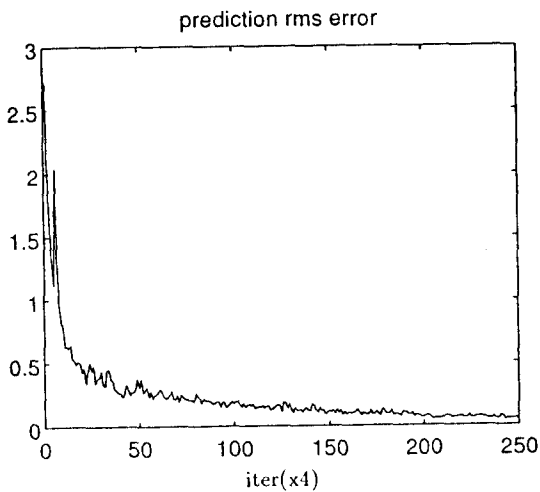


그림 5: 학습횟수에 대한 예측 rms 오차