

대각귀환 신경망을 이용한 비선형 적응 제어

류동완 · 이영석 · 서보혁

· 경북대학교 전기공학과

Adaptive Control of the Nonlinear Systems Using Diagonal Recurrent Neural Networks

Dong-Wan Ryoo · Young-Seog Lee · Bo-Hyeok Seo

Department of Electrical Engineering Kyungpook National University

Abstract - This paper presents a stable learning algorithm for diagonal recurrent neural network(DRNN). DRNN is applied to a problem of controlling nonlinear dynamical systems. A architecture of DRNN is a modified model of the Recurrent Neural Network(RNN) with one hidden layer and the hidden layer is comprised of self-recurrent neurons. DRNN has considerably fewer weights than RNN. Since there is no interlinks amongs in the hidden layer. DRNN is dynamic mapping and is better suited for dynamical systems than static forward neural network. To guarantee convergence and for faster learning, an adaptive learning rate is developed by using Lyapunov function. The ability and effectiveness of identifying and controlling a nonlinear dynamic system using the proposed algorithm is demonstrated by computer simulation .

1. 서론

비선형동적 시스템(nonlinear dynamic system)을 제어하기 위해서 신경망을 이용한 제어기설계가 많이 연구되어 왔다.[1][2] Narendra 등[2]은 기존의 모델 기준 적응 제어와 유사한 구조를 갖는 정적 신경망을 이용하여 비선형 시스템에 대해 제어방법을 보였다.

Chen [3]은 비선형 시스템의 자기 동조 제어를 위한 정적 정방향 신경망의 학습방법을 보였다.

이러한 비선형 동적 시스템의 제어기 설계에 이용되어 온 신경망은 두가지로 분류될 수 있다. 하나는 정적 정방향 신경망(static forward neural network)구조이며, 다른 하나는 동적 귀환 신경망(dynamic recurrent neural network)이다.

정적 정방향 신경망은 정적 시스템(Dynamic system mapping)을 표현하지 않는 정적 맵핑(static mapping)인 반면에 동적 귀환 신경망은 동적 맵핑(Dynamic mapping)이다. 이 중에 동적 시스템을 제어하기에 적합한 신경망은 내부적으로 동적인 특성을 갖는 동적 귀환 신경망이다.[4][5] 이러한 귀환 신경망은 안정성(stability)과 학습 알고리즘(learning algorithm)개발의 두가지 방향으로 연구가 이루어져 왔다.[4][5][6][7]

시스템의 식별기와 제어기의 파라메타가 너무 많아서는 좋지 않으며 또한 대부분의 제어에서 실시간 구현은 중요하다. 제어를 위한 신경망의 연결강도수는 줄여야 할 필요가 있다. 그러나 귀환 신경망은 연결 강도의 수가 많다. 그러므로 온라인 적응 제어를 성공적으로 수행하기 위해서는 무엇보다도 학습속도가 빠른 신경망구조가 필요하며 또한 안정성이 고려되어야 한다.

따라서 본 논문에서는 대각 귀환 신경망 구조를 사용하며, 안정성이 고려된 학습률(learning rate)를 제안한다. 학습률(learning rate)의유도를 위해 리야푸노프 함수를 사용한다.

대각 귀환 신경망의 목적은 동적 귀환 신경망(Recurrent Neural Network : RNN) [6][7] 에 비해 학습시간이 짧다는 점이다. 즉 중간층 뉴런 상호간의 연결이 없기 때문에 동적 귀환 신경망(RNN)에 비해 연결강도의 수가 적다.

또한 제안된 학습 알고리즘을 사용한 대각 귀환 신경망으로 비선형 동적 시스템의 식별과 제어 모의실험을 통해 학습알고리즘의 안정성과 효율성을 확인한다.

2. 정적 신경망과 귀환 신경망의 구조

정적 정방향 신경망의 뉴런은 단순히 가중치가 가해진 입력의 합을 활성화 함수를 통해 출력을 이끌어 낸다. 반면 동적 귀환 신경망의 뉴런은 상태 변수의 상태방정식이 정의됨으로 출력을 이끌어낸다. 정적 정방향 신경망의 수학적인 표현은 식(2-1)과 같다.

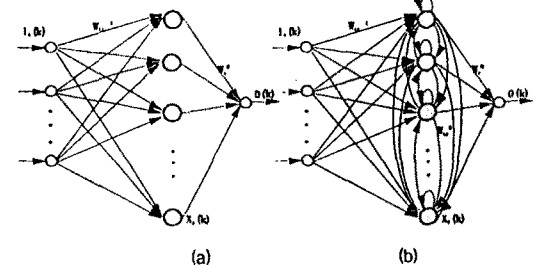
$$O^F(k) = \sum_{j=1}^n W_j^O f(S_j(k)) = \sum_{j=1}^n W_j^O f\left(\sum_{i=1}^m W_{ij}^I I_i(k)\right) \quad (2-1)$$

동적 귀환 신경망의 수학적인 표현은 식(2-2)와 같다.

$$O^H(k) = \sum_{j=1}^n W_j^O f(S_j(k)) = \sum_{j=1}^n W_j^O f\left(\sum_{i=1}^m W_{ij}^I I_i(k) + \sum_{q=1}^n W_{jq}^H X_q(k-1)\right) \\ = \sum_{j=1}^n W_j^O f\left(\sum_{i=1}^m W_{ij}^I I_i(k) + \sum_{q=1}^n W_{jq}^H \cdot f\left(\sum_{i=1}^m W_{iq}^I I_i(k-1) + \sum_{a=1}^n W_{aq}^H X_a(k-2)\right)\right) \quad (2-2)$$

여기서 $X_q(k)$ ($q=1, \dots, n$)는 뉴런의 현재 상태이며, $I_i(k)$ 는 뉴런으로 들어오는 외부입력이며, $S_j(k)$ 는 외부 입력의 합이며, $O(k)$ 는 신경망의 출력이며, W^I, W^H, W^O 는 입력층, 중간층, 출력층의 연결강도이다. $f(\cdot)$ 는 뉴런의 활성화 함수로 하이퍼볼릭 탄젠트(hyperbolic tangent : $\tanh(\cdot)$) 함수를 일반적으로 사용한다. 잊기는 귀환 신경망이 동적 맵핑임을 나타내고 있다.

그림1은 이러한 신경망의 기본구조를 나타낸 것이다.



[그림 1] 정적 정방향 신경망과 동적 귀환 신경망의 구조
(a) 정적 정방향 신경망 (Feedforward Neural Network)
(b) 동적 귀환 신경망 (Recurrent Neural Network)

3. 대각 귀환 신경망제어의 구조와 학습법칙

3.1 대각 귀환 신경망의 구조

대각 귀환 신경망은 귀환신경망에서 변화된 구조를 가진다. 즉 자기 귀환으로 구성된 뉴런으로 구성되어 연결강도의 수가 적다. 수학적인 표현은 식(3-1)과 같다.

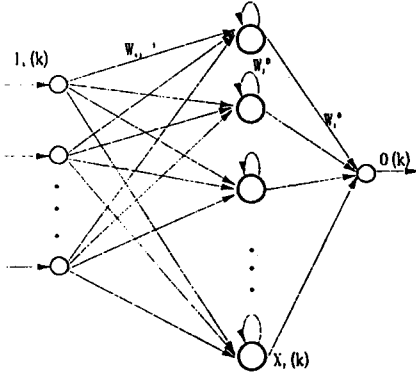
$$O(k) = \sum_j W_j^0 X_j(k) \quad (3-1)$$

여기서, $X_j(k) = f(S_j(k))$

$$S_j(k) = W_j^D X_j(k-1) + \sum_i W_{ij}^I I_i(k) \text{ 이다.}$$

$$\text{또는 } O(k) = \sum_{j=1}^m W_j^0 f \left(\sum_{i=1}^m W_{ij}^I I_i(k) + W_j^D \right) \\ f \left(\sum_{i=1}^m W_{ij}^I I_i(k-1) + W_j^D X_j(k-2) \right) \quad (3-2)$$

그림2 는 대각 귀환 신경망의 구조를 나타내고 있다.



[그림 2] 대각 귀환 신경망의 구조 (DRNN)
대각 귀환 신경망으로 구성된 식별기 수학적인 모델은 식(3-3)과 같다.

$$\hat{y}(k) = \sum_j W_j^0(k) X_j'(k) + \sum_j W_j^0(k) X_j^e(k) U(k) \quad (3-3)$$

$$X_j'(k) = f(S_j'(k)) \quad (3-4)$$

$$S_j'(k) = W_j^D X_j'(k-1) + \sum_i W_{ij}^I I_i(k) \quad (3-5)$$

$$X_j^e(k) = f(S_j^e(k)) \quad (3-6)$$

$$S_j^e(k) = W_j^D X_j^e(k-1) + \sum_i W_{ij}^I I_i(k) \quad (3-7)$$

3.2 비선형 동적 시스템의 제어 알고리즘

식(3-8)은 비선형 동적 시스템 (Discretization of nonlinear dynamics) [3][7][8]을 나타낸다.

$$y(k+1) = f [y(k), y(k-1), \dots, y(k-n), \\ u(k-1), u(k-2), \dots, u(k-m)] \\ + g [y(k), y(k-1), \dots, y(k-n), \\ u(k-1), u(k-2), \dots, u(k-m)] \cdot u(k) \quad (3-8)$$

여기서 $y(k)$ 는 출력이며 $f(\cdot)$ 와 $g(\cdot)$ 는 비선형 함수이다. 오차 함수는 다음과 같이 실제 플랜트의 출력과 식별기의 출력으로 구성된다.

$$E = \frac{1}{2} (y(k) - \hat{y}(k))^2 \quad (3-9)$$

신경망으로 구성된 식별기를 식(3-10)과 같이 나타낼 수 있으며

$$\hat{y}(k+1) = \hat{f}(I, W^I) + \hat{g}(I, W^e) U(k) \quad (3-10)$$

제어입력은 다음과 같이 계산된다.

$$U(k) = \frac{y_d(k+1) - \hat{f}(I, W^I)}{\hat{g}(I, W^e)} \quad (3-11)$$

여기서 $y_d(k+1)$ 는 원하는 출력이다. 신경망의 각 연결강도 오차 기울기는 식(3-12)와 같이 표현된다.

$$\frac{\partial E}{\partial W} = -e(k) \frac{\partial \hat{y}(k)}{\partial W} \quad (3-12)$$

여기서, $e(k) = y(k) - \hat{y}(k)$ 이다.

신경망의 연결강도의 학습 법칙은 다음과 같다.

$$W(n+1) = W(n) + \eta \left(-\frac{\partial E}{\partial W} \right) \quad (3-13)$$

여기서 η 는 학습률(learning rate)이다.

대각 귀환 신경망 $\hat{f}(I, W^I)$ 의 입력층, 중간층, 출력층의 연결강도 (W^{00}, W^{0D}, W^{0I})에 대한 출력의 기울기는 식(3-14)-(3-16)과 같다.

$$\frac{\partial \hat{y}(k)}{\partial W_j^0} = X_j'(k) \quad (3-14)$$

$$\frac{\partial \hat{y}(k)}{\partial W_j^D} = W_j^0 P_j'(k) \quad (3-15)$$

$$\frac{\partial \hat{y}(k)}{\partial W_{ij}^I} = W_j^0 Q_{ij}'(k) \quad (3-16)$$

여기서 $P_j'(k) = \partial X_j'(k) / \partial W_j^D$, $Q_{ij}'(k) = \partial X_j'(k) / \partial W_{ij}^I$ 이며 다음식을 만족한다.

$$P_j'(k) = f'(S_j') (X_j'(k-1) + W_j^D P_j'(k-1)), \\ P_j'(0) = 0 \quad (3-17)$$

$$Q_{ij}'(k) = f'(S_j') (I_i'(k) + W_j^D Q_{ij}'(k-1)), \\ Q_{ij}'(0) = 0 \quad (3-18)$$

대각 귀환 신경망 $\hat{g}(I, W^e)$ 의 입력층, 중간층, 출력층의 연결강도 (W^{e0}, W^{eD}, W^{eI})에 대한 출력의 기울기는 식(3-19)-(3-21)과 같다.

$$\frac{\partial \hat{y}(k)}{\partial W_j^{e0}} = X_j^e(k) U(k) \quad (3-19)$$

$$\frac{\partial \hat{y}(k)}{\partial W_j^{eD}} = W_j^{e0} P_j^e(k) \quad (3-20)$$

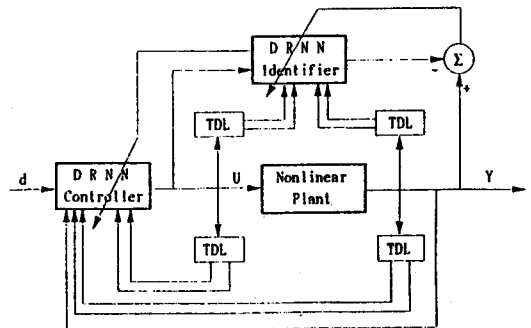
$$\frac{\partial \hat{y}(k)}{\partial W_{ij}^{eI}} = W_j^{e0} U(k) Q_{ij}^e(k) \quad (3-21)$$

여기서 $P_j^e(k) = \partial X_j^e(k) / \partial W_j^{eD}$, $Q_{ij}^e(k) = \partial X_j^e(k) / \partial W_{ij}^{eI}$ 이며 다음식을 만족한다.

$$P_j^e(k) = f'(S_j^e) (X_j^e(k-1) + W_j^{eD} P_j^e(k-1)), \\ P_j^e(0) = 0 \quad (3-22)$$

$$Q_{ij}^e(k) = f'(S_j^e) (I_i^e(k) + W_j^{eD} Q_{ij}^e(k-1)), \\ Q_{ij}^e(0) = 0 \quad (3-23)$$

그림3 은 대각 귀환 신경망으로 구성된 제어기의 블록선도를 나타낸 것이다.



[그림 3] 대각 귀환 신경망으로 구성된 제어기의 블록선도

4. 안정성을 고려한 학습률

식(3-13)의 학습법칙에서 적당한 학습률 η 를 결정하는것은 중요하다. 작은 값의 η 는 수렴성은 보장할지라도 학습속도가 매우 느리게 된다. 반면에 η 가 너무 크면 알고리즘이 불안정하게 된다. 이 절에서는 적당한 학습률 η 를 유도한다. Lyapunov 함수는 다음과 같이 주어진다.

$$V(k) = \frac{1}{2} e^2(k) \quad (4-1)$$

여기서 $e(k)$ 는 학습시 $e(k) = y(k) - \hat{y}(k)$ 이다.

Lyapunov 함수의 변화량은 다음과 같이 나타낼수 있다.

$$\Delta V(k) = V(k+1) - V(k) = \frac{1}{2} [e^2(k+1) - e^2(k)] \quad (4-2)$$

식(4-1)에서의 학습에 의한 오차의 차는 아래와 같이 표현된다.

$$e(k+1) = e(k) + \Delta e(k) = e(k) + \left[\frac{\partial e(k)}{\partial W} \right]^T \Delta W \quad (4-3)$$

여기서 ΔW 는 임의의 연결강도의 변화물 나타낸다. 식(3-10)과 식(3-11)으로부터 식(4-4)를 나타낼수 있다.

$$\Delta W = -\eta e(k) \frac{\partial e(k)}{\partial W} = \eta e(k) \frac{\partial \hat{y}(k)}{\partial W} \quad (4-4)$$

여기서 W 와 η 는 신경망의 임의의 연결강도와 학습률을 나타낸다.

식 (4-2)- (4-4) 로 부터 $\Delta V(k)$ 는 아래와 같이 들 수 있다.

$$\begin{aligned} \Delta V(k) &= \Delta e(k) \left[e(k) + \frac{1}{2} \Delta e(k) \right] \\ &= \left[\frac{\partial e(k)}{\partial W} \right]^T \eta e(k) \frac{\partial \hat{y}(k)}{\partial W} \\ &\quad \cdot \left(e(k) + \frac{1}{2} \left[\frac{\partial e(k)}{\partial W} \right]^T \eta e(k) \frac{\partial \hat{y}(k)}{\partial W} \right) \quad (4-5) \end{aligned}$$

또 $\frac{\partial e(k)}{\partial W} = -\frac{\partial \hat{y}(k)}{\partial W}$ 이므로 식(4-6)을 얻을 수 있다.

$$\begin{aligned} \Delta V(k) &= -\eta e^2(k) \left\| \frac{\partial \hat{y}(k)}{\partial W} \right\|^2 + \frac{1}{2} \eta^2 e^2(k) \left\| \frac{\partial \hat{y}(k)}{\partial W} \right\|^4 \\ &= -\lambda e^2(k) \quad (4-6) \end{aligned}$$

여기서 $G_{MAX} := \max_k \|G(k)\|$, $G(k) = \frac{\partial \hat{y}(k)}{\partial W}$ 그리고

$$\eta_V = \eta G_{MAX}^2 \text{ 라 두면}$$

$$\begin{aligned} \lambda &= \frac{1}{2} \|G(k)\|^2 \eta (2 - \eta \|G(k)\|) \\ &= \frac{1}{2} \|G(k)\|^2 \eta \left(2 - \frac{\eta_V \|G(k)\|^2}{G_{MAX}^2} \right) \\ &\geq \frac{1}{2} \|G(k)\|^2 \eta (2 - \eta_V) > 0 \quad (4-7) \end{aligned}$$

식(4-7)로 부터 $0 < \eta_V < 2$ 를 얻을 수 있다. 그러므로 η 가 식(4-8)을 만족하면 학습 알고리즘의 수렴성을 보장한다.

$$0 < \eta < \frac{2}{G_{MAX}^2} \quad (4-8)$$

여기서 $\|\cdot\|$ 는 Euclidean norm이다.

$$\text{식(4-7)에서 } \eta (2 - \eta_V) > 0 \quad (4-9)$$

또는 $\eta_V (2 - \eta_V) / G_{MAX}^2 > 0$ 이다

여기서 $\eta_V = \eta G_{MAX}^2$ 이므로 안정성이 고려되고, 빠른 수렴을 보장하는 학습률의 값은 $\eta_V = 1$ 일때이다. 즉

$$\eta^* = \frac{1}{G_{MAX}^2}$$

η_j^0 , η_j^D , 와 η_j^I 를 각각의 신경망 연결강도 W^{j0} , W^{jD} , 그리고 W^{jI} 의 학습률이라고 할 때 η_j^0 , η_j^D , 그리고 η_j^I 는 다음과 같다. 여기서 $0 < |W_j^{jD}| < 1$

$$j=1, 2, \dots, h_j \text{ 이다.}$$

$$(a) \quad 0 < \eta_j^0 < \frac{2}{h_j} \quad (4-10)$$

$$(b) \quad 0 < \eta_j^D < \frac{2}{h_j} \left[\frac{1}{W_{MAX}^{j0}} \right]^2 \quad (4-11)$$

$$(c) \quad 0 < \eta_j^I < \frac{2}{(n_j + h_j)} \left[\frac{1}{W_{MAX}^{j0} I_{MAX}^j} \right]^2 \quad (4-12)$$

여기서 h_j 는 $\hat{f}(\cdot)$ 중간층의 뉴런수이고, n_j 는 신경

망의 입력 갯수이며,

$$W_{MAX}^{j0} := \max_k \|W^{j0}(k)\|, \quad I_{MAX}^j := \max_k \|I^j(k)\|$$

그리고 $\|\cdot\|$ 는 sup-norm 이다.

η_j^0 , η_j^D , 와 η_j^I 를 각각의 신경망 연결강도 W^{j0} , W^{jD} , 그리고 W^{jI} 의 학습률이라고 할때 η_j^0 , η_j^D , 그리고 η_j^I 는 다음과 같다. 여기서 $0 < |W_j^{jD}| < 1$

$$j=1, 2, \dots, h_j \text{ 이다}$$

$$(a) \quad 0 < \eta_j^0 < \frac{2}{U_{MAX}^2 h_j} \quad (4-13)$$

$$(b) \quad 0 < \eta_j^D < \frac{2}{U_{MAX}^2 h_j} \left[\frac{1}{W_{MAX}^{j0}} \right]^2 \quad (4-14)$$

$$(c) \quad 0 < \eta_j^I < \frac{2}{U_{MAX}^2 (n_j + h_j)} \left[\frac{1}{W_{MAX}^{j0} I_{MAX}^j} \right] \quad (4-15)$$

여기서 h_j 는 $\hat{g}(\cdot)$ 중간층의 뉴런수이며, n_j 는 신경망의 입력 갯수이며,

$$W_{MAX}^{j0} := \max_k \|W^{j0}(k)\|, \quad I_{MAX}^j := \max_k \|I^j(k)\|$$

$U_{MAX} = \max_k |U(k)|$ 그리고 $\|\cdot\|$ 는 sup-norm 이다.

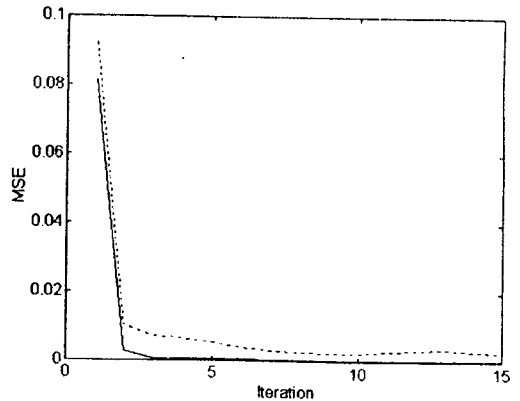
5. 사례연구와 검토

제안된 제어 알고리즘들 다음과 같은 비선형 시스템 모델 [3][6][7] 에 적용하여 제어 성능을 조사하였다.

$$y(k+1) = a \sin[2y(k)] + b u(k)$$

a=0.8 b=1.2

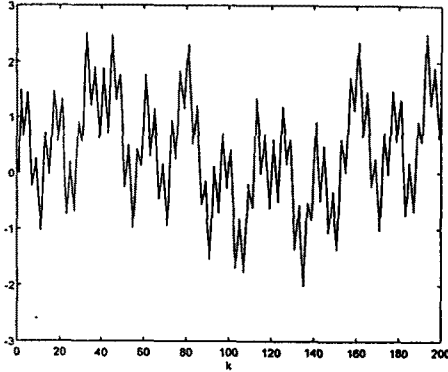
$$u(k) = 0.2 [\sin(0.5 r_k) + \cos(2 r_k) + \sin^2(3 r_k) + \cos^3(4 r_k) \\ \cos(5 r_k) + \sin(20 r_k)] \quad \text{and} \quad r_k = \pi k / 40$$



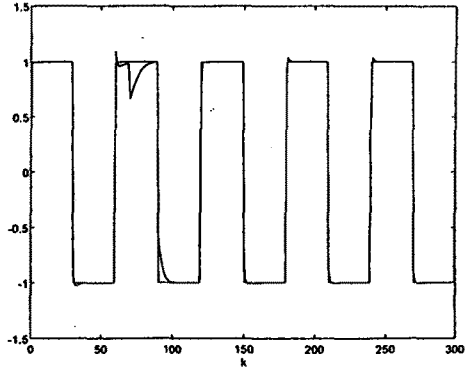
[그림 4] 오차(Mean Squared Error)의 비교

DRNN : solid line RNN : dashed line
그림 4 에서 RNN[6][7]에 비해 연결강도의 수를 줄임에도 불구하고 안정성이 고려된 학습률을 사용하여 학습시킨 결과 거의 같은 성능을 보이며 오차의 수렴도 안정적이다.

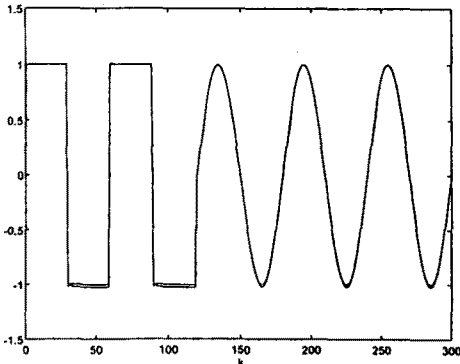
RNN의 학습률은 [7]과 같은 값을 이용하였고, DRNN의 학습률은 본 논문에서 제시된 안정성이 고려된 적응 학습률을 사용하였다.



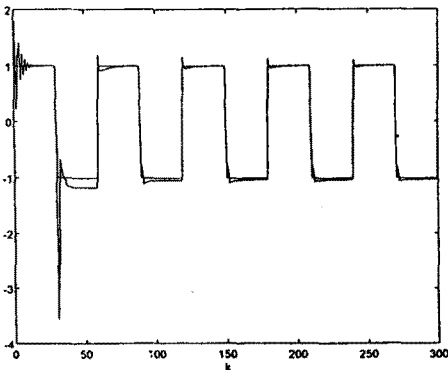
[그림 5] 200 쌍(pairs) 를 15회 학습 후 시스템 출력(y)과 식별기(\hat{y}) 출력
 $y(k)$: solid line $\hat{y}(k)$; dashed line



[그림 8] 시스템 파라메타 a 와 b 가 70 번째 샘플링 순간에 a=0.8 에서 a=0.6 , b=1.2 에서 b=0.8 로 변했을 때의 성능 (15회 학습 후)



[그림 6] 원하는 출력과 시스템 출력 (15회 학습 후)



[그림 7] 연결강도(Weighting)의 학습없이 랜덤 초기화 후 온 라인 제어

6. 결 론

동적 시스템을 제어하기에 적합한 신경망은 내부적으로 동적인 특성을 갖는 동적 귀환 신경망이다. 본 논문에서는 식별기의 학습속도와 제어기 입력의 계산속도를 줄이기 위해 동적 귀환 신경망의 연결강도수를 간략하게 한 대가 귀환 신경망을 사용하므로써 RNN[7]에 비해 학습속도가 빠르며, 또한 연결강도의 수를 줄임에도 불구하고 안정성이 고려된 학습률을 사용하여 학습시킨 결과 거의 같은 성능을 보이며 오차의 수렴도 안정적임을 보였다.

또한 제안된 학습 알고리즘을 사용한 대가 귀환 신경망으로 비선형 시스템의 식별과 제어 모의실험을 통해 학습알고리즘의 안정성과 효율성을 확인하였다.

앞으로의 연구 방향으로는 부정확한 식별기 신경망의 정보를 가지고 학습된 제어기신경망에 해당하는 제어오차에 대한 제어 입력을 구하는 문제를 고려해 보고, 미지의 시스템을 제어할 때 축차된 식별기와 제어기 신경망에 대한 안정도에 대해 살펴보는 것이다.

참 고 문 헌

- [1] Jeffrey L. Elman, Neural Networks for Control, The MIT Press, 1990.
- [2] K.S. Narendra and K. Pathasathary, "Identification and control of dynamical systems using neural networks," IEEE Trans. on Neural Networks, vol 1, pp. 4-27, 1990.
- [3] C. F. Chen, " Back-propagation neural networks for nonlinear self-tuning adaptive control," IEEE Contr. Syst. Mag, vol. 10 . no 3, pp. 44-48 ,Apr. 1990.
- [4] Liang Jin, Peter N. Nikiforuk, and Madan M. Gupta, " Dynamics and Stability of Mutilayered Recurrent Neural Networks," in Proc. 1993 IEEE Int. Conf Neural Net, 1993 Vol II, pp. 1135-1140.
- [5] George A. Rovithakis, and A. Christodoulou, "Adaptive Control of Unknown Plants Using Dynamical Neural Networks," IEEE Trans on SMC, Vol 24 No.3 pp.400-411, march ,1994.
- [6] S.I. Sudharsanan and M. K. Sundareshan, "Training of a three layer dynamical recurrent neural network for nonlinear input-output mapping," in Proc. Int. Joint Conf. Neural Networks (IJCNN-91), vol II, Seattle, WA 1991, pp.111-116
- [7] A. Karakasoglu, S.I. Suharsanan, and M.K. Sundareshan "Identification and Decentralized Adaptive Control Using Dynamical Neural Networks with Application to Robotic Manipulators," IEEE Trans on Neural Networks, vol.4, no 6, pp.919-930, 1993
- [8] K.S. Narendra and J. Balakrishnan, and M.K. Ciliz "Adaptation and Learning Using Multiple Models, Switching, and Tuning" IEEE Contr. Syst. Mag vol15 no3, pp37-51 June 1995 Trans on
- [9] M.M Polycarpou and P.A. Ioannou, "Learning convergence analysis of neural-type structured networks," IEEE Trans. Neural Networks, vol. 3, no. 1 pp.39-50 Jan.1992.