

神經回路網을 利用한 非線形 시스템의 Indirect Adaptive制御에 관한 研究

The study on the Indirect Adaptive Control of
Nonlinear System using Neural Network

*金 成 柱 , 李 尙 培

韓國海洋大學校 電子通信工學科

*Sung Joo Kim, Sang Bae Lee

Dept. of Electro. & Comm. Eng, Korea Maritime Univ.

Abstract

In this paper, we demonstrate that neural networks can be used effectively for the control of nonlinear dynamical system. To adaptively control a plant, there are two distinct approach. these are direct control and indirect control. Both direct and Indirect adaptive control are trained using static back propagation. In indirect, using the resulting identification model, which contains neural networks and linear dynamical elements as subsystems, the parameters of the controller are adjusted.

I. 서 론

학습되지 않은 신경망은 마치 갓난 아기나 길들여지지 않은 동물과 같아서 입력을 주었을 때 어떤 반응을 할지는 아무도 예측할 수 없다. 뿐만 아니라 똑같은 신경망이라도 그것에 무엇을 학습 시키냐에 따라 신경망의 동작은 완전히 달라진다. 신경망에게 지능을 부여하는 즉, 학습시키는 과정은 일련의 자료들을 경험하도록 하는 것을 통해 이루어진다. 그렇게 하면 신경망은 주어진 학습 규칙에 따라 스스로 weight를 조절하여, 주어진 자료들로부터 자기 나름대로 어떠한 지식을 습득하게 된다. 신경망의 학습 규칙에는 여러 가지가 있으나 일반적으로 많이 사용하는 Back Propagation 알고리즘을 사용하여 비선형 시스템의 제어에 이용하였다.

이 논문은 신경망이 비선형시스템의 Control에 효과적으로 사용될 수 있음을 예증하고, 그것의 적용 제어 능력에 대해서 서술하였다. 어떠한 Controller의 제어 능력은 얼마나 우리가 원하는 출력을 낼 수 있도록 제어할 수 있느냐에 달려 있다. 그래서 이러한 제어 시스템에 신경망의 학습 능력을 이용하여 Controller를 구성하였을 때, 응답 특성을 고찰하여 보았으며, 제어 대상이 되는 Plant를 알 수 없는 경우에도 신경망을 이용하여 미지의 Plant를 식별(Identification)하고, 이것을 바라는 출력을 낼 수 있도록 제어하는 것을 곁하였다.

한편 본 논문의 구성은 II절에서는 앞에서 소개한 Back Propagation 알고리즘에 대해서 나타내었고, III절과 IV절에서는 직접 적용 제어와 간접 적용 제어에 대해서 다루었고, 각각의 장단점에 대해서 V절에서 비교, 설명을 하고 있다.

II. 신경회로망의 학습 알고리즘(Back Propagation)

Back Propagation 알고리즘에는 multilayer로 구성되는 것과 recurrent network를 이용한, 즉 Static과 Dynamic의 두 가지가 있으나, 여기서는 본 논문에서 사용한 Static에 대해서만 다루었다. 그림 1.은 hidden1, hidden2, 그리고 output의 3 layer network를 나타내고 있으며, 각 layer에 따른 weight를 w_i^1 , w_j^2 , w_k^3 로 나타내었다.

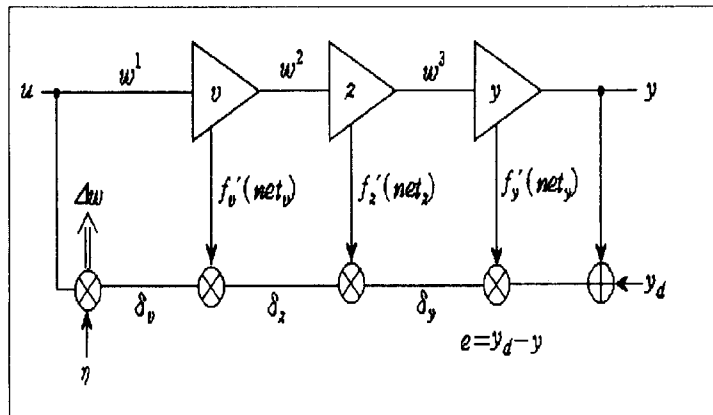


그림 1. 다층 신경회로망의 기본 구조.

weight의 변화률을 구하기 위해서 다음의 3단계의 수학적 모델링으로 요약할 수 있다.

(1) 각layer에서의 출력을 구한다.

hidden1에서의 출력

$$net_v = \sum_{i=0}^{N_v} w_i^1 u_i, \quad v = 1, 2, \dots, N_v \quad (1)$$

$$f_v(net_v) = \frac{1}{1 + \exp(-net_v)} \quad (2)$$

hidden2에서의 출력

$$net_z = \sum_{j=0}^{N_j} w_j^2 f_v(net_v) , \quad z = 1, 2, \dots, N_z \quad (3)$$

$$f_z(net_z) = \frac{1}{1 + \exp(-net_z)} \quad (4)$$

output에서의 출력

$$net_y = \sum_{k=0}^{N_k} w_k^3 f_z(net_z) , \quad y = 1, 2, \dots, N_y \quad (5)$$

$$y = net_y \quad (6)$$

(2) error항의 역전파

일단, 신경회로망이 주어진 입력에 대해 동작하고 나면, 출력층 neuron들의 error가 구해진다. 출력층 neuron들의 경우 error를 구하는 방법은 바라는 값에서 출력 neuron의 활성값을 뺀 값이 바로 출력층 neuron들의 error가 된다. 단층(one layer)의 경우, 이렇게 구한 error를 사용하여 바로 weight를 조절하지만, 다층(multi layer)에서는 위와 같은 방법을 사용할 수 없다.

그 이유는 hidden층 neuron의 경우, 출력층 neuron와는 달리 바라는 값으로부터 직접 error를 구할 수 없기 때문이다. 그래서 이 error를 한번더 가공하여 각각의 출력층 neuron에 대해 델타(δ)를 구하여 이것을 역전파시켜 각 layer에서의 weight를 조절한다.

$$E = \frac{1}{2}(y_d - y)^2 \quad (7)$$

$$\frac{\partial}{\partial y_{output}} E = -(y_d - y) \quad \text{그리고,} \quad (8)$$

$$f_y(net_y)' = \frac{\partial y}{\partial (net_y)}, \quad y_d - y = -\frac{\partial}{\partial y} E = \frac{\partial E}{f_y'(net_y) \partial (net_y)} \quad (9)$$

$$\delta_y \equiv -\frac{\partial}{\partial (net_y)} E = (y_d - y) f_y'(net_y) \quad (10)$$

$$\delta_z \equiv -\frac{\partial}{\partial (net_z)} E = -\frac{\partial E}{\partial z} \frac{\partial z}{\partial (net_z)} \quad (11)$$

$$\begin{aligned} \text{여기서, } \frac{\partial E}{\partial z} &= \frac{\partial}{\partial z} \left[\frac{1}{2} (y_d - f_y'(net_y))^2 \right] \\ &= -(y_d - y) \frac{\partial}{\partial z} \{f_y'(net_y)\} \\ &= -(y_d - y) f_y'(net_y) \frac{\partial (net_y)}{\partial z} \\ &= -\delta_y w_k^3 \end{aligned} \quad (12)$$

그러므로

$$\delta_z = w_k^3 \delta_y f_z'(net_z) \quad (13)$$

$$\delta_v = -\frac{\partial E}{\partial (net_v)} = -\frac{\partial E}{\partial v} \frac{\partial v}{\partial (net_v)} \quad (14)$$

여기서

$$\begin{aligned}
 \frac{\partial E}{\partial v} &= \frac{\partial}{\partial v} \left[\frac{1}{2} \{y_d - f_y(\text{net}_y)\}^2 \right] \\
 &= -(y_d - y) \frac{\partial}{\partial v} \{f_y(\text{net}_y)\} \\
 &= -(y_d - y) f_y'(\text{net}_y) \frac{\partial}{\partial v} (\text{net}_y) \\
 &= -(y_d - y) f_y'(\text{net}_y) \frac{\partial}{\partial v} (w_k^3 f_z(f_v \text{net}_v w_j^2)) \\
 &= -(y_d - y) f_y'(\text{net}_y) w_k^3 f_z'(f_v \text{net}_v w_j^2) \frac{\partial}{\partial v} (f_v \text{net}_v w_j^2) \\
 &= -w_j^2 \delta_z
 \end{aligned} \tag{15}$$

그러므로

$$\delta_v = w_j^2 \delta_z f_v'(\text{net}_v) \tag{16}$$

그래서 *weight*의 변화율은

$$\Delta w_i^1 = \eta \delta_v u_i \tag{17}$$

(3) *weight*의 조절

각 층에 있어서의 *weight*변화율은 아래와 같다.

$$\begin{aligned}
 w_k^{3'} &= w_k^3 + \eta \delta_y f_z(\text{net}_z) \\
 w_j^{2'} &= w_j^2 + \eta \delta_z f_v(\text{net}_v) \\
 w_i^{1'} &= w_i^1 + \eta \delta_v u_i
 \end{aligned} \tag{18}$$

이상의 3단계에 의해서 신경회로망은 주어진 자료에 의해 자신의 *weight*를 변화시킬 수 있다.

이러한 내부의 *weight*조절에 의해서 우리가 원하는 출력을 내어 주는 것이다.

Simulation에 사용된 활성화함수는

$$f(x) = \frac{1 - \exp(-\lambda \text{net})}{1 + \exp(-\lambda \text{net})} \text{ 이다.} \tag{19}$$

그리고, error항에 이용되는 $f(x)'$ 는 다음과 같이 유도 될 수 있다.

$$\begin{aligned}
 f(\text{net}) &= \text{output} = \frac{1 - \exp(-\lambda \text{net})}{1 + \exp(-\lambda \text{net})} \\
 \text{여기서, } \exp(-\lambda \text{net}) &= \frac{1 - \text{output}}{1 + \text{output}} \\
 f(\text{net})' &= \frac{2\lambda \exp(-\lambda \text{net})}{1 + \exp(-\lambda \text{net})^2} = \frac{1}{2} \lambda (1 - \text{output})(1 + \text{output}) \\
 f(\text{net})' &= \frac{1}{2} \lambda (1 - f(\text{net}))(1 + f(\text{net}))
 \end{aligned} \tag{20}$$

III 신경회로망을 이용한 Direct Adaptive Control

여기서 사용되는 Direct Adaptive Controller의 구성도는 다음과 같다.

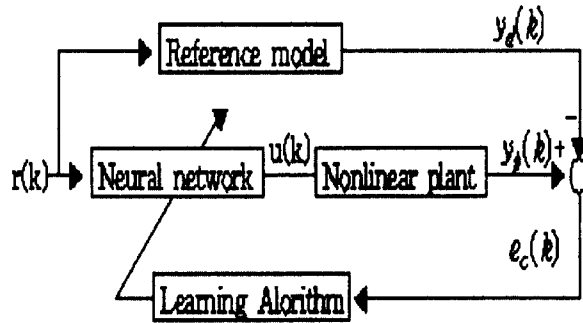


그림.2. 신경회로망을 사용하는 비선형 플랜트의 직접 적응 제어 시스템

직접 적응 제어 방식에서 Controller의 변수들은 출력error가 줄어들도록 직접적으로 조정되는 방식이다. neural network의 출력이 Plant의 Control의 입력이 되며, 그것의 입력은 Plant의 지연된 입력과 출력이 되고 있다. 여기서 두개의 플랜트에 대해서 Simulation을 해보았다.

Ex 1)

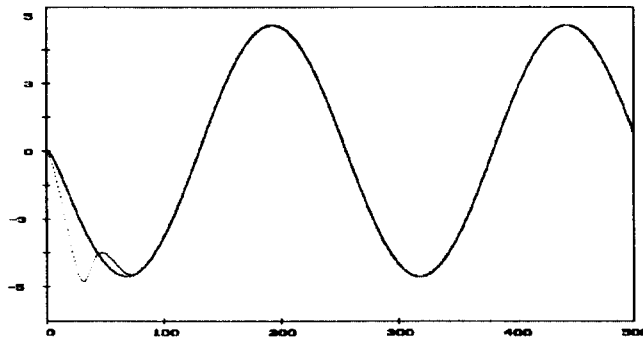
$$\text{입력 } r(k) = \sin(2\pi k/250), \quad -1 \leq r(k) \leq 1,$$

$$f[] = r(k)$$

$$\text{Plant Model} \quad : \quad y_p(k+1) = f[] + u(k)$$

$$\text{Reference Model} \quad : \quad y_m(k+1) = 0.6y_m(k) + 0.2y_m(k-1) + r(k)$$

에 대한 시뮬레이션 결과는 다음과 같다.



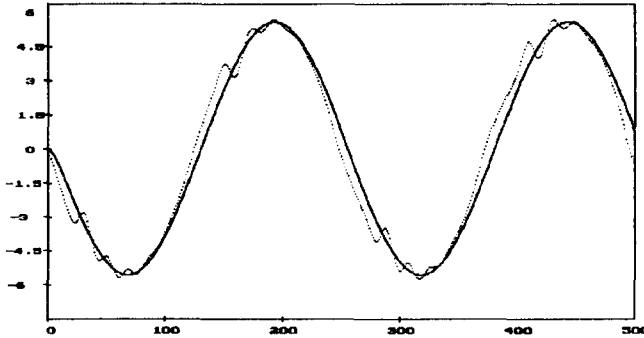
Ex 2)

$$r(k) = \sin(2\pi k/250), \quad -1 \leq r(k) \leq 1$$

$$f[\] = 0.5r(k) + 0.1\cos(2\pi k/250)$$

$$\text{Plant Model} : y_p(k+1) = f[\] + u(k)$$

$$\text{Reference Model} : y_m(k+1) = 0.6y_m(k) + 0.2y_m(k-1) + r(k)$$



미지의 플랜트에 대해서는 현재까지 출력error에 기반을 둔 control의 변수를 직접적으로 제어하는 방법은 유효하지 않기 때문에 간접 적응 제어를 위한 신경회로망 시스템을 새로 모델링 하였다. 그 내용은 다음절에서 설명된다.

IV 신경회로망을 이용한 Indirect Adaptive Control의 구성

간접 적응 방식의 시스템의 구성도를 그림3에서 나타내고 있다.

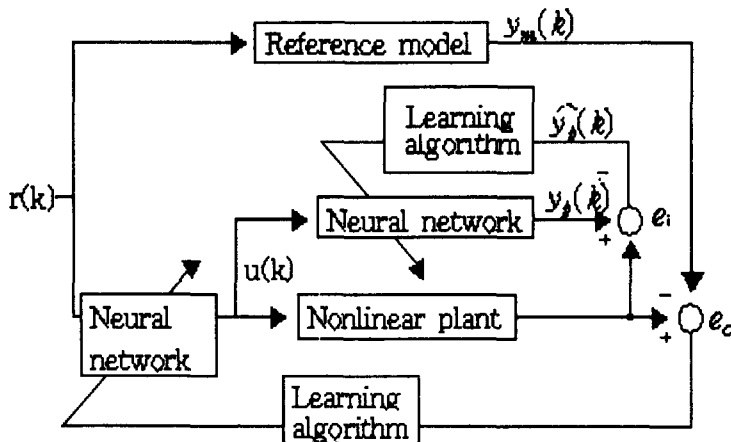


그림.3. 신경회로망을 사용하는 간접 적응 제어 시스템

다음의 Ex3과 Ex4는 간접 적응 방법으로 Control을 행하여 직접 적응 제어 방법보다 효과적이라는 것을 보여주기 위하여 Ex1과 Ex2에서 사용한 Reference Model과 Plant Model을 그대로 사용하여 비교,검토하였다. 그 결과 직접 적응 방법이 Control시간이나 적은 Iteration 으로 만족스러운 결과를 얻을 수 있으나, 이것이 제대로 적용되지 않는 경우에는 간접 적응 제어 방법으로 원하는 응답 특성을 얻을 수 있다는 것을 이 논문에서 보여준다.

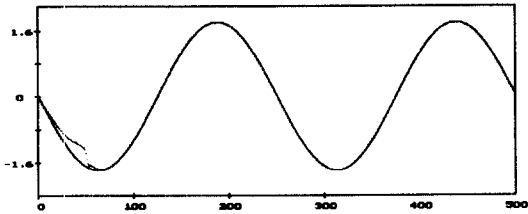
Ex3)

$$\text{입력 } r(k) = \sin(2\pi k/250), \quad -1 \leq r(k) \leq 1,$$

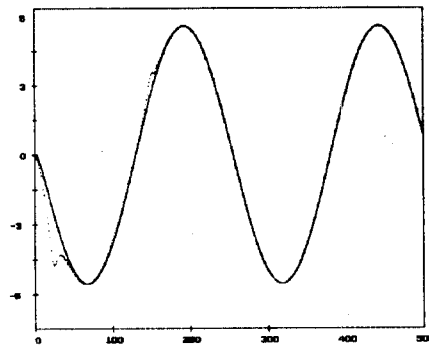
$$f[\] = r(k)$$

$$\text{Plant Model} \quad : y_p(k+1) = f[\] + u(k)$$

$$\text{Reference Model} \quad : y_m(k+1) = 0.6y_m(k) + 0.2y_m(k-1) + r(k)$$



Ex3의 Identification



Ex3의 Control

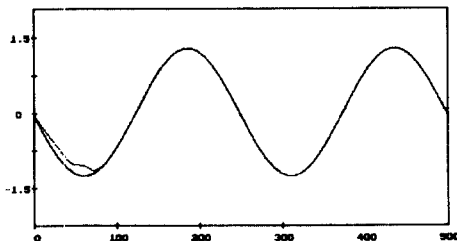
Ex4)

$$r(k) = \sin(2\pi k/250), \quad -1 \leq r(k) \leq 1$$

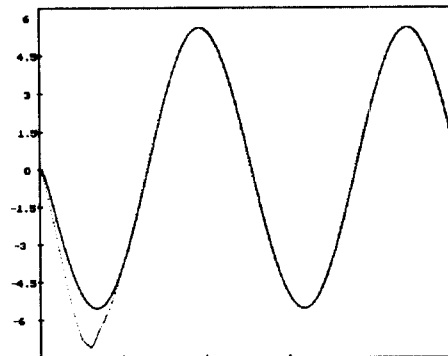
$$f[\] = 0.5r(k) + 0.1\cos(2\pi k/250)$$

$$\text{Plant Model} \quad : y_p(k+1) = f[\] + u(k)$$

$$\text{Reference Model} \quad : y_m(k+1) = 0.6y_m(k) + 0.2y_m(k-1) + r(k)$$



Ex4의 Identification



Ex4의 Control

그리고, 간접 적응 제어 방식이 미지의 플랜트에 대해서 효과적인 제어를 할 수 있다는 것을 보여주기 위하여 다음과 같은 예제를 실행시켜 보았다. 여기서 플랜트의 함수를 알지못하므로 $f[]$ 는 neural network와 series-parallel 방법을 사용하여 $f[]$ 로써 식별해 내어야 한다.

이렇게 Identification과정이 끝나고 난뒤, $y_m - y_p = e_c = 0$ 가 되도록 Control입력 u 를 구해서 제어하는 방법이 간접 적응 제어 방식이다.

Ex 5)

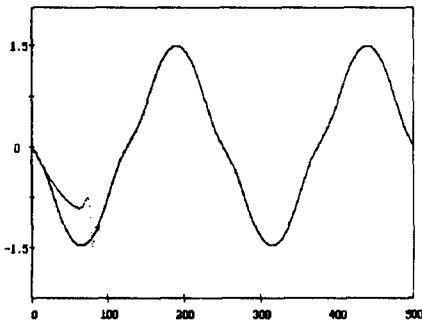
$$r(k) = \sin(2\pi k/250), -1 \leq r(k) \leq 1$$

$$f[y_p(k), y_p(k-1)] = \frac{y_p(k)^2 y_p(k-1)}{1 + y_p(k)^2 + y_p(k-1)^2}$$

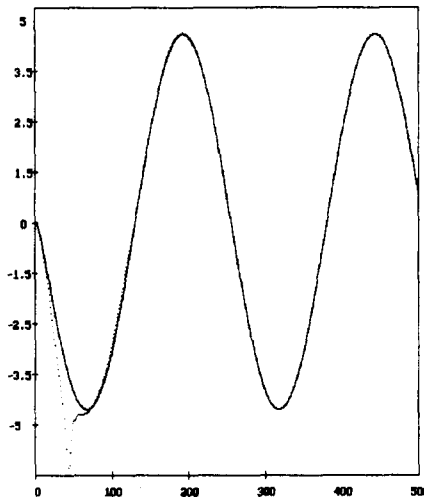
Plant Model : $y_p(k+1) = f[] + u$

Reference Model : $y_m(k+1) = 0.6y_m(k) + 0.2y_m(k-1) + r(k)$

에 대한 시뮬레이션 결과는 다음과 같다.



Ex5의 Identification



Ex5의 Control

V 결 론

플랜트를 모르는 경우, 위에서 언급 한대로 미지의 플랜트를 Identification 하고 그 이후 Control을 해야 하기 때문에 Indirect adaptive Control방식만이 제어에 유효하다. 그리고 플랜트를 알고 있는 경우에도 Direct adaptive Control방식보다도 Indirect adaptive Control방식이 플랜트를 나타내는 방정식의 차수에 융통성이 있음을 알 수 있었다.

즉 이것은 우리가 원하는 값으로의 제어수용폭이 훨씬더 넓음을 말해 주고 있다. 즉, 다시 말하면 한번의 학습 과정만 하는 Direct adaptive Control방식이 제어할 수 있는 플랜트의 범위가 한정적인 반면에 Indirect adaptive Control방식은 Direct adaptive Control방식보다 그것이 넓다는 것이다.

또한 높은 차수의 Nonlinear multivariable systems의 제어에도 Indirect adaptive Control방식이 효과적으로 사용될 수 있다는 것을 알았다.

참고문헌

- [1] Jacek M. Zurada, *Introduction to Artificial Neural Systems.*, West Publishing Company, pp. 163-235, 1992.
- [2] Jerry M. Mendel, *A Prelude to Neural Networks : Adaptive and Learning System.*, Englewood Cliffs, NJ: PTR Prentice Hall, 209-317, 1994.
- [3] K. S. Narendra and K. Parthasarathy, "*Neural networks and dynamical systems.* Part II: Identification," Center for Syst. Sci., Dept. of Electrical Eng., Yale University, New Haven, CT, tech. rep. 8902, Feb. 1989.
- [4] K. S. Narendra and K. Parthasarathy, "*Neural networks and dynamical systems.* Part III: Control," Center Syst. Sci., Dept. Electrical Eng., Yale University, New Haven CT, tech. rep. 8902, May. 1989.
- [5] K. G. shin and X. Cui ., "*Design of a Knowledge-Based Controller for Intelligent Control System.*" IEEE Trans. Sys., Man, Cybern., vol. 21, no. 2, pp. 368-375, 1991.