

# 신경망을 이용한 적응제어기의 추적 성능 평가

## (Tracking Performance Evaluation of Adaptive Controller using Neural Networks)

°최수열\*, 박재형\*\*, 한창훈\*\*, 박선국\*\*\*

\* 서일전문대학 전자통신과 (Tel:+82-02-433-0015)

\*\* 동아대학교 전자공학과 (Tel:+82-051-200-7704)

\*\*\* 경남전문대학 전자과 (Tel:+82-051-320-1341)

**Abstract** In the study, Simulation result was studied by connecting PID Controller in series to the established Neural Networks Controller. Neural Network model is composed of two layers to evaluate tracking performance improvement. The regular dynamic characteristics was also studied for the expected error to be minimized by using Widrow-Hoff delta rule. As a result of the study, We identified that tracking performance improvement was developed more in case of connecting PID than Neural Network Controller and that tracking plant parameter in 251 sample was approached rapidly in case of time variable.

**Keyword** Neural network, Widrow-Hoff, learning rule, Backpropogation

### 1. 서론

오늘날 제어이론의 발달과 더불어 마이크로프로세서가 대량 보급됨에 따라 제어 시스템의 최적 조정법과 자동 동조법 등이 연구되고 있으나, 여기에도 많은 문제점이 수반되고 있다. 대부분 제어 시스템은 여러 가지 기능을 갖고 있어 특성이 복잡해 지고 주위의 변화에도 민감한 특성을 갖고 있다. 이를 극복하기 위해 주위의 변화에 덜 민감하고 추적성능이 빠른 제어 시스템 설계가 요구되고 있다. 이러한 요구에 부합된 연구를 위해 그동안 많은 연구진들이 노력해 왔는데 그 중의 하나가 적응제어<sup>1,2</sup>라 할 수 있다. 이것은 시스템의 파라미터 변화에 대처 할 수 있는 장점이 있으나, 계산시간이 많이 소요되며 경우에 따라서 커다란 외란에 의해 추적 성능이 저하되는 단점을 가지고 있다<sup>3</sup>.

이러한 문제점을 해결하기 위해 신경망 제어 시스템의 이론적 배경을 토대로 한 연구가 활발히 진행되고 있다<sup>4,5,6,7</sup>.

1995년 Zhang<sup>8</sup> 등은 산업용 추적 시스템에 적합한 적응신경 제어기를 온라인으로 학습하는 방법을 제안하였다.

본 논문에서는 Zhang등이 제안한 온라인 학습방법을 이산 시스템에 적용하여 추적 성능을 개선하고자 한다. 우선 이산 시스템에 적합한 2계층 신경 모델을 설정하여 플랜트의 출력과 신경망에 의해 예측한 출력의 차를 최소화 되도록 시스템의 정규적인 동특성을 학습하였다. 이 학습방법은 Widrow-Hoff델타 규칙에 적용하여 2계층 신경망의 가중치를 조절하였으며 성능 지수를 최적화 시키는 제어 신호를 구하였다.

그리고 추적성능을 개선하기 위해 산업현장에 많이 사용하고 있는 PID(Proportional Integral Derivative)제어기를 신경망과 직렬로 구성하여 성능을 분석하였으며, 추적할 플랜트의 파라미터가 시변인 경우에도 추적성능이 개선됨을 확인하였다.

### 2. 플랜트의 신경망 모델 구성

#### 2.1 문제 정식화

선형 DARMA(Deterministic Autoregressive Moving-Average)모델의 전달함수는 다음처럼 표현된다.

$$A(q^{-1})y(k) = B(q^{-1})u(k) \quad (1)$$

여기서  $A(q^{-1})$ 과  $B(q^{-1})$ 의 다항식은

$$\begin{aligned} A(q^{-1}) &= 1 + a_1q^{-1} + \dots + a_{n_a}q^{-n_a} \\ B(q^{-1}) &= b_1q^{-1} + \dots + b_{n_b}q^{-n_b} \end{aligned} \quad (2)$$

이며,  $q^{-1}$ 은 후향 연산자 ( $q^{-1}y(k) = y(k-1)$ ).  $u(k), y(k)$ 는 플랜트 입출력을 나타낸다.

플랜트 모델을 설정하기 위해 다음과 같은 세가지 가정을 전제로 하고 있다.

**가정 1:** 차수의 상한선  $n_a, n_b$ 는 기지이다.

**가정 2:**  $B(q^{-1})$ 은 안정한 다항식이다.

**가정 3:** 계수  $b_i \neq 0$

이러한 가정은 적응제어법칙[1,2]에서도 적용된다. 따라서 가정 1은 추정될 실제 시스템의 차수이며, 플랜트의 신경망모델과 제어기는  $n, m(=n_a, n_b)$ 로 한다. 가정 2는 신경망에 대해 완전한 추적과 페루프 안정도를 얻기 위함이고 가정 3은 제어기가 순차적으로 수행됨을 의미한다.

#### 2.2 플랜트의 신경망 모델

본 논문의 신경망 모델은 문제 정식화에 의해서 선형 2계층으로 구성하였다. 그림 1에서 처럼 신경망 모델의 입력계층은  $n+m$ 개의 노드로 구성되었으며, 이러한  $n+m$ 입력은 플랜트에서 이미 측정된 입출력신호이다. 그리고 신경망의 출력층은 한 개의 노드만 가지며 플랜트의 출력을 예측한다. 따라서 그림 1에서 처럼 플랜트 모델인 신경망 추정자는 Widrow-Hoff델타 규칙을 적용하여 플랜트 동특성을 학습시킨다.

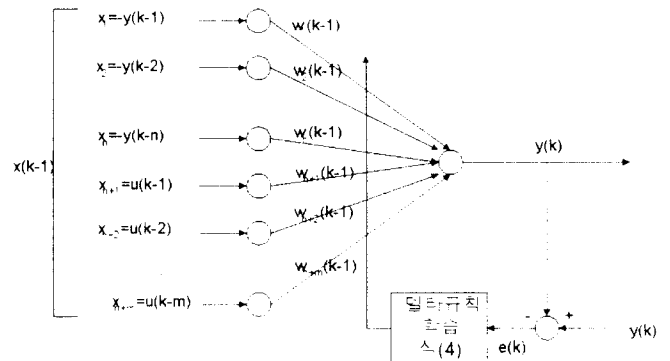


그림 1. 플랜트의 신경망 모델  
Fig. 1. Neural model of the plant

시간  $k$ 에서 가중치 벡터는 다음과 같이 정의하고 있다.

$$w(k) = [w_1(k), \dots, w_n(k), w_{n-1}(k), \dots, w_{n-m}(k)]^T \quad (3)$$

그리고 가중치는 Widrow-Hoff규칙을 이용하여 다음과 같이 갱신한다.

$$w(k+1) = w(k) + \frac{\alpha e(k+1)x(k)}{\varepsilon + x^T(k)x(k)} \quad (4)$$

여기서  $\alpha \in (0, 2)$ 는 학습률 높이기 위한 감소인자<sup>9</sup>이며,  $x^T(k)x(k)=0$ 인 되는 경우를 피하기 위해 임의의 상수  $\varepsilon$ 을 추가하였다. 이 상수는 거의 영에 가까운 작은 값으로 선정한다

여기서  $x(k)$ 는 망입력벡터이며, 이 경우 다음과 같이 정의된다.

$$x(k) = [x_1(k), \dots, x_n(k), x_{n+1}(k), \dots, x_{n+m}(k)]^T \quad (5)$$

$$= [-y(k), \dots, -y(k-na+1), u(k), \dots, u(k-nb+1)]$$

이 그림에서 나타난 것처럼 오차신호  $e(k)$ 는 플랜트 응답  $y(k)$ 와 신경 모델에 의해 예측된 응답  $\hat{y}(k)$ 의 차로 나타나며, 이 값은 다음과 같이 표현된다.

$$\hat{y}(k) = \sum_{i=1}^{n+m} w_i(k-1)x_i(k-1) \quad (6)$$

따라서 적응 알고리즘(식(4))은 오차신호를 최소화하는 것이다<sup>[5]</sup>.

플랜트의 동특성 정보는 신경추정자의 결합가중치를 포함하고 있다. 이러한 가중치를 이용하여 식 (6)처럼 플랜트 출력을 예측할 수 있다.

### 3. 신경망제어기 구성

#### 3.1 기존의 신경제어기

위에서 언급한 신경 추정자는 플랜트 동특성의 학습을 통하여 가중치를 조절하고 제어신호  $u(k)$ 를 발생하게 된다. 따라서 이 제어 신호가 플랜트 입력에 가해질 경우 플랜트 출력  $y(k)$ 는 기준신호  $r(k)$ 가 됨을 의미한다.

델타규칙을 통해 직접 제어되는 기존신경제어기는 그림 2와 같이 2계층 신경망으로써  $n+m$ 개의 노드를 갖는 입력층과 한 개의 출력층으로 구성되어 있다.

여기서 시간  $k$ 의 입력벡터는 다음과 같이 정의된다.

$$z(k) = [r(k+1), -x_1(k), -x_2(k), \dots, -x_n(k) - x_{n+2}(k), \dots, -x_{n+m}(k)]^T \quad (7)$$

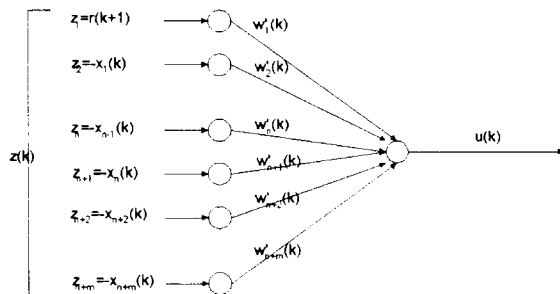


그림 2. 신경제어기  
Fig. 2. Neural Control

결합가중치벡터  $\hat{w}(k)$ 는 다음과 같이 플랜트의 신경모델에 가중치  $w_i, i=1, \dots, n+m$ 을 조절할 수 있는 함수로 정의된다.

$$\hat{w}(k) = \frac{1}{w_{n+1}(k)} [1, w_1(k), w_2(k), \dots, w_n(k), w_{n+2}(k), \dots, w_{n+m}(k)]^T \quad (8)$$

따라서 식 (8)-(9)를 신경제어기에 적용하여 성능지수가 최적화되는 제어법칙을 구한다. 따라서 이 그림에 나타난 것처럼 제어신호  $u(k)$ 는 입력벡터  $z(k)$ 의 일차결합이며 새로운 가중치에 따라서  $\hat{w}_i(k)$ 가 발생된다.

$$u(k) = \sum_{i=1}^{n+m} \hat{w}_i(k)z_i(k) \quad (9)$$

그러므로 전반적인 페루프 추적 제어 시스템은 그림 3으로 나타낸다.

#### 3.2 PID 신경 제어기

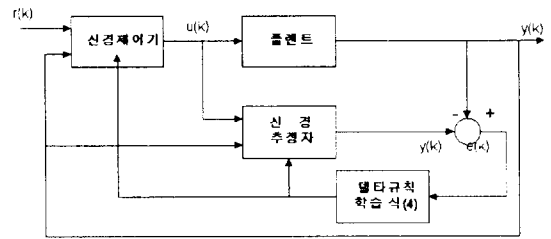


그림 3. 적응제어기 구조  
Fig. 3. Adaptive control scheme

추적 성능을 개선하기 위해 그림 4와 같이 신경망제어기에 고전적인 PID제어기를 직렬 연결하였다. 이 때 플랜트의 동특성 학습은 2장에서 제시한 Widrow-Hoff델타 규칙으로 학습하였다.

이 때 제어기 이득을 적절하게 자동적으로 조절함으로써 추적 성능을 개선할 수 있다.

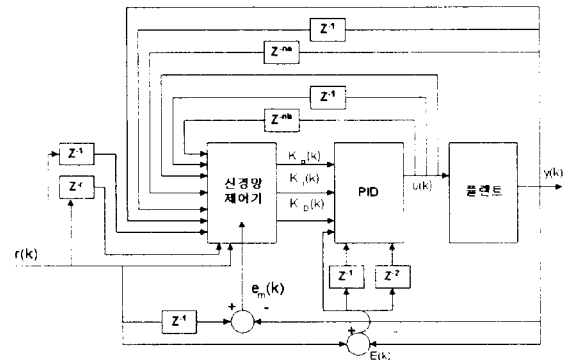


그림 4. 신경제어기의 적응PID  
Fig4. Adaptive PID of neural controller

한편 PID신경망 제어기는 PID이득을 출력한 신경망과 이 이득을 사용한 제어 입력을 연산하는 노드로 구성한다.

그림 4에서 나타난 PID제어기는 종래의 PID이득 계산식으로부터 다음과 같이 얻어진다.

$$u(k) = u(k-1) + (K_P(k) + K_I(k) + K_D(k))E(k) - (K_P(k) + 2K_D(k))E(k-1) + K_D(k)E(k-2) \quad (10)$$

단,  $K_P(k) = C_1 O_P(k), K_I(k) = C_1 O_I(k), K_D(k) = C_1 O_D(k), E(k) = r(k) - y(k)$ 이다.

여기서  $u(k)$ 는 제어입력이고  $K_P(k), K_I(k), K_D(k)$ 는 비례, 적분, 미분의 이득,  $O_P(k)$ 는  $O_I(k), O_D(k)$ 신경망제어기의 출력이다.  $C_1$ 은 스케일 조정 정수이고  $r(k)$ 는 기준신호.  $y(k)$ 는 플랜트 출력이다.

그림 4의 신경망 제어기는 3계층으로 구성하였다. 본 논문에서는 요구하는 응답  $r(k)$ 에 대한 입력층의 노드 수를 한 개로 하고, 중간층(은닉층)의 노드수를 세 개로 한다. 그리고 PID이득을 출력하는 출력층 노드수를 3개로 한다.

신경망 제어기의 학습은 역전파법으로 행하고 평가함수  $J$ 를 최소화 하도록한다.

$$J = e_m(k)^2 / 2 \quad (11)$$

여기서,  $e_m(k) = C_2 [r(k-1) - y(k)]$ 이고,  $C_2$ 은 스케일 조정 정수이다.

이와 같은 경우 신경망제어기의 가중치를 수정하는 데 사용하는 평가함수  $J$ 를 다시 쓰면 다음과 같다.

$$\frac{\partial J}{\partial w_{ij}(k-1)} = e_m(k) \times \frac{\partial e_m(k)}{\partial w_{ij}(k-1)} = -C_2 e_m(k) \frac{\partial y(k)}{\partial w_{ij}(k-1)} \quad (12)$$

여기서,  $w_{ij}(k-1)$ 은 시각  $k-1$ 에 대한  $J$ 번째 노드와  $I$ 번째 노드와의 결합가중치를 나타낸다.

$$\frac{\partial y(k)}{\partial w_{ij}(k-1)} = \frac{\partial y(k)}{\partial u(k-1)} \times \frac{\partial u(k-1)}{\partial w_{ij}(k-1)} \quad (13)$$

식 (13)의 제 1항은 입출력의 비를 결정하고 차분 및 신경망 모델의 역전파법에 의해 구해질 수 있다. 식 (13)의 제 2항은 다음 식처럼 구한다.

$$\frac{\partial u(k-1)}{\partial w_{ij}(k-1)} = \frac{\partial u(k-1)}{\partial K(k-1)} \times \frac{\partial K(k-1)}{\partial w_{ij}(k-1)} \quad (14)$$

여기서,  $K(k-1)$ 은 신경망 제어기의 출력이며  $K(k-1) = [K_p(k-1), K_i(k-1), K_D(k-1)]^T$ 가 된다. 식(14)의 제2항은

$$\frac{\partial K(k-1)}{\partial w_{ij}(k-1)} = \left[ \frac{\partial K_p(k-1)}{\partial w_{ij}(k-1)}, \frac{\partial K_i(k-1)}{\partial w_{ij}(k-1)}, \frac{\partial K_D(k-1)}{\partial w_{ij}(k-1)} \right]^T \quad (15)$$

된다. 식(15)는 신경망제어기에 대해 역전파법을 이용하여 쉽게 구할 수 있다.

식 (14)의 제 1항

$$\frac{\partial u(k-1)}{\partial K(k-1)} = [u_{a1}(k-1), u_{a2}(k-1), u_{a3}(k-1)]$$

은 다음 식을 사용하여 계산된다.

$$\begin{aligned} u_{a1}(k-1) &= \frac{\partial u(k-1)}{\partial K_p(k-1)} = E(k-1) - E(k-2) \\ u_{a2}(k-1) &= \frac{\partial u(k-1)}{\partial K_i(k-1)} = E(k-1) \\ u_{a3}(k-1) &= \frac{\partial u(k-1)}{\partial K_D(k-1)} = E(k-1) - 2E(k-2) + E(k-3) \end{aligned} \quad (16)$$

식 (10)-(16)으로부터 평가함수를 신경망제어기에 적용한 가중치는 다음과 같다.

$$\begin{aligned} \frac{\partial J}{\partial w_{ij}(k-1)} &= -C_2[r(k-1) - y(k)] \frac{\partial y(k)}{\partial u(k-1)} \\ &\left[ u_{a1}(k-1) \frac{\partial K_p(k-1)}{\partial w_{ij}(k-1)} + u_{a2}(k-1) \frac{\partial K_i(k-1)}{\partial w_{ij}(k-1)} + \right. \\ &\left. u_{a3}(k-1) \frac{\partial K_D(k-1)}{\partial w_{ij}(k-1)} \right] \end{aligned} \quad (17)$$

여기서,  $\frac{\partial J}{\partial w_{ij}(k-1)}$ 를 순차적으로 가중치를 갱신하고 그 식은 다음과 같다.

$$\begin{aligned} \Delta w_{ij}(k-1) &= -\epsilon \frac{\partial J}{\partial w_{ij}(k-1)} + \alpha \Delta w_{ij}(k-2) + \beta \Delta w_{ij}(k-3) \\ w_{ij}(k-1) &= w_{ij}(k-1) + \Delta w_{ij}(k-1) \end{aligned} \quad (19)$$

여기서,  $\epsilon$ 는 학습률을 나타내며  $\alpha$ 는 모멘텀항이고  $\beta$ 는 제어감쇄로 나타낸다. 따라서  $k$ 순간에서 제어루프의 순차적인 동작은 다음과 같다.

- $r(k)$ 과  $y(k)$  측정
- 과거 가중치  $w_{ij}(k-1)$ 을 이용하여 출력  $y(k)$ 를 예측을 계산하기 위해 플랜트의 신경모델 사용
- 오차신호  $e(k) = y(k) - \hat{y}(k)$ 를 계산하고 새로운 가중치  $w_{ij}(k)$ 를 델타 규칙 적용
- 식(8), 식(10)을 이용하여 신경 제어기 가중치  $w_{ij}(k)$ 를 갱신
- 최적화되는 제어 신호를 발생하기 위해 기존신경 제어기와 PID신경제어기를 사용한다.

#### 4. 시뮬레이션결과 및 비교 검토

본 논문에서 제안한 적응 신경망 제어의 추적 성능 평가는 시뮬레이션을 통하여 입증한다.

##### 4.1 제어 대상 모델링

본 논문에서 제어대상 모델링은 다음과 같이 불안정한 시스템으로 모델링한다.

$$y(k) = 0.55y(k-1) + 0.46y(k-2) + 0.07y(k-3) + 0.1u(k-1) + 0.02u(k-2) + 0.03u(k-3) + e(k)$$

본 연구의 시뮬레이션을 위해 플랜트 파라미터를 일정한 값으로 하였고 각 변수 및 크기 값은 다음과 같이 선정하였다

- 요구하는 기준입력은 크기가  $\pm 1$ 인 계단파형으로 선정하였다. 또한 신경추정자의 초기 가중치 벡터는

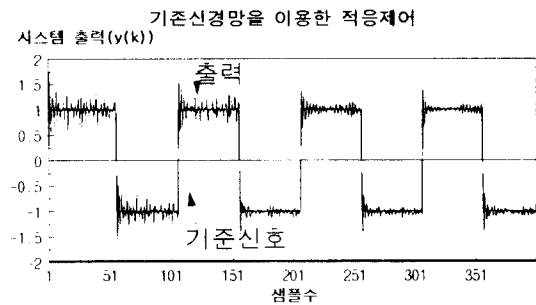
$$u[0] = [0, 0, 0, 1, 0, 0]^T$$

로 하였으며, PID신경망의 초기 가중치는  $i=j$ 일 때  $w_{ij} = -0.02$ 로,  $i \neq j$  경우  $w_{ij} = 0.01$ 로 선정하였다. 그 외의 모든 초기 값은 영으로 가정하였고, 신경망 추정자의 델타 규칙 식 (4)에서  $\alpha = 1$ ,  $\epsilon = 0.001$ 으로 선정하였다. 그리고 식(18)에서  $\alpha = 0.04$ ,  $\beta = 0.9$ ,  $\epsilon = -0.1$ 로 하였다.

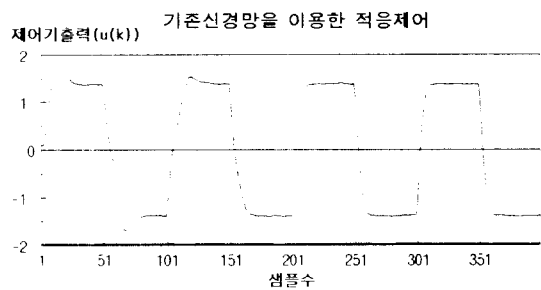
- 시스템 잡음은 평균값이 영으로하고 잡음 공분산은 요구되는 출력의  $\pm 1\%$  범위내로 가정하여 0.01로 하였다.

##### 4.2 비교 검토

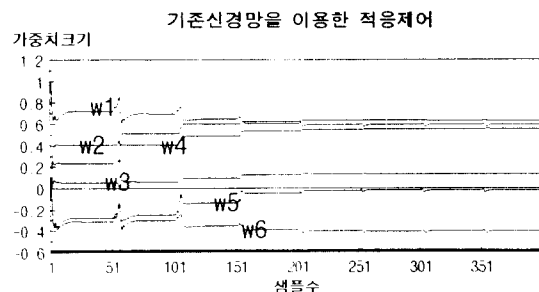
그림 5(a)는 플랜트에 잡음을 가한 경우 시스템의 특성을 나타내며 이 그림에 나타난 것처럼 적응과도 후에는 시스템 출력이 요구되는 기준 신호에 매우 잘 추적됨을 알 수 있다. 그림 (b)는 플랜트의 입력인데 신호의 크기를  $\pm 1$ 으로 설정하였다. 그림 (c)는 신경망 추정자로 나타내었다.



(a) 시스템 출력



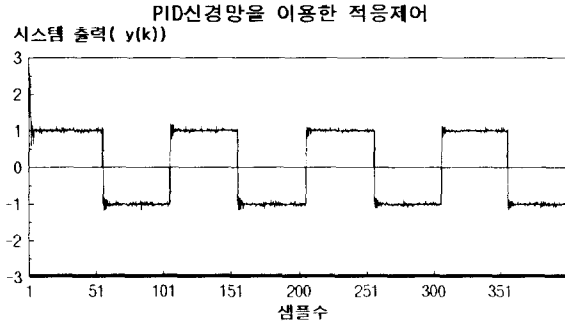
(b) 제어기 출력



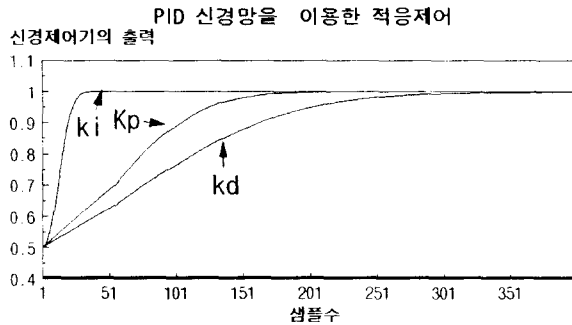
(c)가중치

그림 5 기존신경망을 이용한 적응제어  
Fig. 5 adaptive Neural Network control

그림 6(a)는 PID신경 적용 제어기를 이용한 시스템의 특성을 나타내고 있으며, 그림 (b)는 가중치를 이용한 이득계수를 나타내고 있다. 가중치 출력에 시그모이드함수를 사용하였다. 여기서 그림 5(a)와 그림 6(a)를 고찰하면 그림 5(a)의 경우 첫 번째 시스템 과도현상이 가중치의 변화에 따라서 불규칙하게 발생되고 있는데 비해 그림 6(a)의 경우는 6샘플까지 발생하였다. 따라서 그림 6(a)은 그림 5(a)보다 과도 시간이 크게 짧아지고 추적 현상이 개선됨을 알 수 있다.



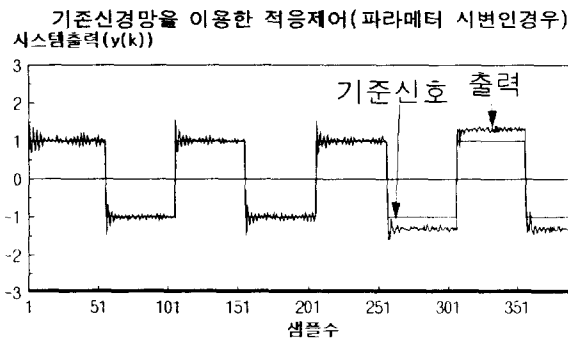
(a)시스템의 출력



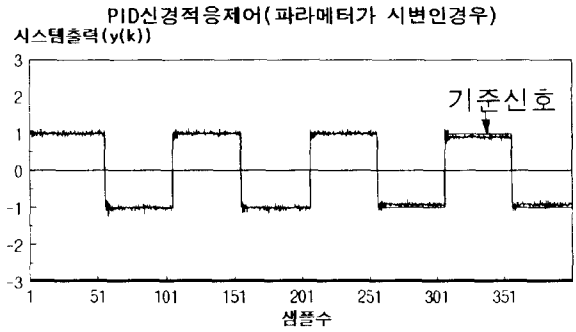
(b)가중치를 이용한 이득 계수

그림 6. PID신경망을 이용한 적응 제어  
Fig. 6. Neural adaptive control of PID

그리고 그림 7(a), (b)는 시스템 파라미터가 251샘플에서  $a_1 = 0.7$ ,  $a_2 = 0.5$ ,  $a_3 = 0.1$ 로 변화했을 때 시스템의 출력을 나타낸다. 이 그림에서도 알 수 있듯이 PID신경적용제어가 훨씬 빠르게 추적됨을 알 수 있으며, (b)의 경우는 (a)보다 출력의 변화가 적음을 알 수 있는데 이것은 3계층의 가중치 변화에 기인하기 때문이다.



(a) 기준신경망의 시스템출력



(b) PID신경망의 시스템 출력

그림 7 시스템파라미터가 시변인 경우  
Fig 7. Time varying of system parameter

여 시뮬레이션한 결과 추적 성능이 개선 되었음을 확인했다. Widrow-Hoff델타 규칙을 이용하여 미지의 동특성을 학습하고, 가중치를 이용하여 제어 신호를 온라인으로 발생 시킨 결과 추적 오차를 줄일 수 있었다. 그리고 페루프 시스템은 전반적으로 안정하였으며 제어성능도 개선되었음을 확인하였다. 이 알고리즘은 시스템 잡음이 존재하는 SISO(Single Input Single Output)통계학적 동특성을 제어하는 데 사용될 수 있다. 앞으로의 연구과제는 미지의 기준 입력을 제어기에 포함하여 해석하고 모든 통계학적 조건을 수용할 수 있는 제어알고리즘 개발에 있다.

## 참고문헌

- [1] K.J. Astrom and B. Wittenmark, "On Self tuning regulators", *Automatica*, Vol. 9, pp. 185-199, 1973.
- [2] P.J. Gawthrop, "Self tuning PID Controllers: algorithm and implementation", *IEEE Trans. Automat.*, Vol. ac-31, No.3, pp. 201-209, 1986.
- [3] SIRA-RAMIREZ, J.H., and ZAK, S.H., "The adaption of perceptron with applications to inverse dynamics identification of unknown dynamic system", *IEEE Trans. Syst., Man, Cybern.*, 1991, SMC-21 pp.634-643.
- [4] D. Psaltis and A.Sideris and A.A.Yamamura, "A multilayered neural network controller", *IEEE Control Systems Magazine*, Vol. 8, No. 2, pp. 17-21, 1988.
- [5] D.H.Nguyen and B.Widrow, "Neural networks for self-learning control systems", *IEEE Control Systems Magazine*, Vol. 10, No. 3, pp. 18-23, 1990.
- [6] T.Yamada and T.Yabuta, "An extension of neural network direct controller", *IEEE Int. Workshop on Intelligence Robots and Systems. IROS '90*.
- [7] K.S. Narendra and K. Pathasarathy, "Identification and control of dynamical systems using neural network", *IEEE Trans. on Neural Network*, Vol.1, No.1, pp. 4-27, 1992.
- [8] Yao Zang and Pratyush Sen. and Grant E. Hearn, "An On-Line Trained Adaptive Neural Controller", *IEEE Control Systems Magazine*, Vol.10, No.4, pp.67-75, 1995.
- [9] L.Jin and P.N.Nikiforuk and M.M.Gupta, "Direct adaptive output tracking control using multilayered neural network", *IEEE proceeding-D*, Vol.140, No.6, pp.393-398, 1993.
- [10] F.C.Chen, "Back-propagation Neural networks for nonlinear self tuning adaptive control", *IEEE Control Systems Magazine*, Vol.10, No.2, pp.44-48, 1990.

## 5. 결 론

본 연구에서는 기존의 신경망 제어기에 PID를 직렬 연결하