

동적 신경망에 의한 비선형 시스템 제어

노용기* 유인호** 조현섭*** 오성곤**** 장성환****
 *군산직업훈련원 **익산대학 ***청운대학교 ****원광대학교

Nonlinear System Control for DNP

Yong Gi Roh* . In Ho Ryu** . Hyeon Seob Cho*** . Seong Kwon Oh**** . Seong Whan Jang****
 *Kunsan V T I . **Iksan National College . ***ChungWoon Univ. . ****Wonkwang Univ.

Abstract - The intent of this paper is to describe a neural network structure called dynamic neural processor(DNP), and examine how it can be used in developing a learning scheme for computing robot inverse kinematic transformations. The architecture and learning algorithm of the proposed dynamic neural network structure, the DNP, are described. Computer simulations are demonstrate the effectiveness of the proposed learning using the DNP.

1. 서 론

미지의 변수를 가진 선형 시스템의 식별과 적응제어에 관한 연구는 지난 30년간 꾸준히 연구되어 왔으며 괄목할 만하게 진보되었다. 하지만 기존의 적응제어 기법은 동적 시스템의 수학적 모형을 필요로 하지만 동적 시스템은 비선형성을 가지고 있으며, 비선형 특성이 심한 시스템의 제어가 아주 어렵다. 실제로 동적 시스템의 모형은 비선형성, 불확정성, 시간지연, 시변 시스템의 변수, 구조 등에 의해서 정확하게 기술한다는 것은 불가능하다. [1] 본 논문에서는 이러한 단점을 보완하기 위하여 동적 사상 특성을 추가 시킨 dynamic neural unit(DNU)를 병렬연결한 DNP를 제안하고 신경망의 적응적인 가중치에 대한 학습 알고리즘을 제시한다. 그리고 이를 비선형 동적 프랜트에 적용하여 컴퓨터 시뮬레이션한 결과를 DNU제어기와 DNP제어기와 비교하였다. [2-3]

2. 신경망의 구조

2.1 DNP의 구조와 수학적 모델

적응적인 시냅스 강도를 갖는 DNP의 기본적인 기능은 dynamic neural unit(DNU)에 기초한다. [4-5] (i) Dynamic Neural Unit(DNU)

DNU는 기억 요소와 그림 1과 같이 전환경로 및 궤환 경로 시냅스 가중치로 구성된다. 이 동적 구조의 출력은 시변 비선형 활성화 함수에 대한 인자를 구성한다. DNU는 크게 (i) synaptic operation (ii) somatic operation의 두가지의 동작을 수행한다. 전자는 전환경로 및 궤환경로 시냅스 가중치의 적용에 대응되고 후자는 비선형 활성화 함수의 이득(형태)의 적용에 대응된다. DNU는 각각 비선형 활성화 함수에 이어지는 2차 구조를 나타내는 시냅스 가중치 a_{ff} 와 b_n 에 의해 가중되는 전환경로 및 궤환경로 지연 요소가 구성된다.

$$v_1(k) = -b_1 v_1(k-1) - b_2 v_1(k-2) + a_0 s(k) + a_1 s(k-1) + a_2 s(k-2)$$
 (1)
 이때 $s(k) \in R^n$ 은 신경 입력 벡터이고, $v_1(k) \in R^1$ 은 동적 구조의 출력이며, $u(k) \in R^1$ 은 신경 출력, k 는 이산 시간 지표, z^{-1} 은 단위 지연 연산자, $a_{ff} = [a_0, a_1, a_2]$ 와 $b_n = [b_1, b_2]$ 는 다음과 같이 정의 된다.

$$\Gamma^T(k, v_1, s) = [v_1(k-1) \ v_2(k-1) \ s(k) \ s(k-1) \ s(k-1)]$$
 (2)

$$\xi^T(a_{ff}, b_n) = [-b_1 \ -b_2 \ a_0 \ a_1 \ a_2]$$
 (Γ :transpose) (3)
 (2)와 (3)에 의해 식(1)은 다음과 같이 된다.

$$V_1(k) = \Gamma(k, v-1, s) \xi^T(a_{ff}, b_n)$$
 (4)

$v_1(k)$ 에 대한 비선형 사상은 다음과 같은 출력을 낸다.

$$u(k) = \Psi[g_s v_1(k) - \theta]$$
 (5)

이때 $\Psi[\cdot]$ 는 비선형 활성화 함수로서 보통은 sigmoidal 함수이고 g_s 는 활성화 함수의 기울기를 제어하는 somatic gain이며, θ 는 뉴런을 점화시키기 위한 임계치(threshold)이다. 여기와 금지 입력 모두에 대한 수학적인 동작을 확장하기 위해 $[-1, 1]$ 에 대한 활성화 함수를 다음과 같이 정의한다.

$$\Psi[v(k)] = \tan[g_s v_1(k) - \theta] = \tanh[v(k)]$$
 (6)

이때 $v(k) = g(s)v_1(k)$ 이다.

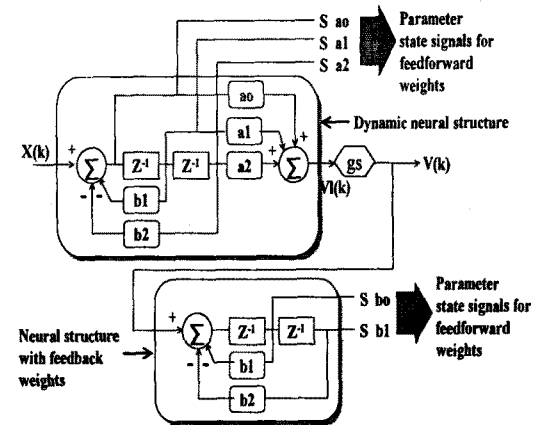


그림 1 DNU 매개변수의 상태신호 모델 (ii) Dynamic Neural Processor(DNP)

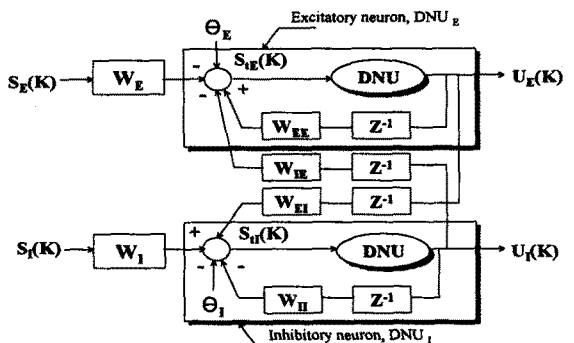


그림 2 DNP의 구조 dynamic neural processor(DNP)는 그림 2에서와 같이 여기와 금지 방식으로 결합된 두 개의 DNU로 구

성된다.

이 구조에서 $s_{\lambda}(k)$ 와 u_{λ} 는 첨자 λ 가 여기 E나 금지 I 상태를 가르킬 때 신경 계산 단위의 자극(입력)과 상태 반응(출력)을 나타낸다. $s_{\lambda}(k)$ 는 신경 단위의 총 입력을 나타내고 $w_{\lambda\lambda}$ 는 한 뉴런에서 다른 뉴런으로의 상호 시냅스 연결 강도(그림2의 w_{IE}, w_{EI})를 나타낸다. 신경 계산 단위인 DNU에 의해 여기되는 기능적 동역학은 식(1)에 나타난 바와 같이 2차 차분방정식으로 정의된다. 제안된 신경 프로세서의 여기 및 금지 신경 단위에 의해 시간(K+1)에 발생하는 상태 변수 $u_E(K+1)$ 와 $u_I(K+1)$ 는 다음과 같이 모델링된다.

$$u_E(k+1) = E[u_E(k), v_E(k)], \text{ and} \\ u_I(k+1) = I[u_I(k), v_I(k)] \quad (7)$$

이때 $v_E(k)$ 와 $v_I(k)$ 는 내부 임계치보다 더 큰 입력을 받아들이는 신경단위에서 뉴런의 비를 나타내고 E와 I는 뉴런의 여기와 금지 작용을 나타낸다. 임계치보다 더 큰 입력을 받아들이는 뉴런은 비선형 함수 $v_{\lambda}(k)$ 로 주어진다. 여기서 금지 신경 단위에 수반되는 총 입력은 각각 다음과 같다.

$$s_{uE}(k) = w_{EE} s_E(k) + w_{EI} u_E(k-1) - \theta_E \quad (8)$$

$$s_{uI}(k) = w_{IS} s_I(k) - w_{II} u_I(k-1) + w_{EI} u_E(k-1) - \theta_I \quad (9)$$

이때 w_E 와 w_I 는 각각 여기와 금지 신경 단위의 비례상수 (scaling factor)이고, w_{EE} 와 w_{II} 는 자기 시냅스 연결 강도를, w_{IE} 와 w_{EI} 는 상호 뉴런 시냅스 연결 강도, θ_E 와 θ_I 는 여기 및 금지 뉴런의 임계치를 각각 나타낸다. 뉴런의 여기와 금지 뉴런의 absolute refractory period(뉴런이 재 점화할 수 없는 기간)를 나타낸다.

$$u_E(k+1) = u_E(k) + (1-r_E)u_E(k) \\ \Psi_E(s_{uE}(k)) : \text{여기 뉴런} \quad (10a)$$

$$u_I(k+1) = u_I(k) + (1-r_I)u_I(k) \\ \Psi_I(s_{uI}(k)) : \text{금지 뉴런} \quad (10b)$$

식 (8)과 (10)에 의해 등복각 곡선에 대한 방정식은 다음과 같이 쓸 수 있다.

$$u_I(k) = \frac{1}{w_{IE}} [(w_E s_E(k) - \theta_E + \\ \Psi_E^{-1} \left[\frac{u_E(k)}{(1-r_E)u_E(k)} \right] + w_{EE} u_E(k))] \text{ for} \\ u_E(k+1) = 0 \quad (11a)$$

$$u_E(k) = \frac{1}{w_{EI}} [(-w_I s_I(k) - \theta_I + \\ \Psi_I^{-1} \left[\frac{u_I(k)}{(1-r_I)u_I(k)} \right] + w_{II} u_I(k))] \text{ for} \\ u_I(k+1) = 0. \quad (11b)$$

sigmoid 함수의 수학적 성질에 의하여, Ψ_E 와 Ψ_I 는 $[-\infty, \infty)$ 의 범위에서 단조 증가하는 유일한 역을 가진다. 식 (11a)로 정의되는 u_I 는 언제나 u_E 의 단조 증가 함수이다. 반면에, 식(11b)에서 Ψ_I^{-1} 앞의 (-) 부호에 의해 u_E 는 점차적으로 감소하는 u_I 의 함수가 될 것이다. 두 등복각 곡선 사이의 이러한 정성적인 차이는 여기와 금지간의 반대칭에 대한 직접적인 결과이다. 신경 집단의 기능적인 동작이 선형 시스템 이론에 의해 근사 될 수 있다는 사실에 기인하여 DNP의 응답 $u(k)$ 는 신경 부차집단에서 여기와 금지의 개별적인 응답인 $u_{\lambda}(k)$ 의 중첩이며 다음으로 주어진다.

$$u(k) = u_E(k) + u_I(k) \quad (12)$$

식 (12)의 의미는 신경 집단의 총체적인 활동이 여기와 금지 후 시냅스 응답의 합이다.

3. DNP 학습 알고리즘 개발

3.1 DNP 제어기 학습 알고리즘

학습 과정은 전향경로 및 퀴환경로 가중치와 오차 함수

를 최소화 하기 위한 somatic gain의 적응과정을 포함한다. 반복 학습 기법에서 제어의 순서는 각 반복 학습 단계에서 목표 상태 $u_d(k)$ 에 도달하기 위한 신경 출력 $u(k)$ 를 발생하기 위해 변형된다. 즉, 초기조건의 임의 집합에 대해 오차 $e(k)$ 와 매개변수 벡터 Ω (aff. bfb. gs. $w_{\lambda\lambda}$)의 성분은 각 학습 횟수 k 와 함께 변한다.

$$u(k) \rightarrow u_d(k) \text{ as } k \rightarrow \infty \text{ or,} \\ \lim_{k \rightarrow \infty} [u_d(k) - u(k) = e(k)] \rightarrow 0 \quad (13)$$

Ω (aff. bfb. gs. $w_{\lambda\lambda}$)(k+1)을 구하기 위해서는 정보집합 $\{e(k-m), e(k), \Omega$ (aff. bfb. gs. $w_{\lambda\lambda}$)(k) $\}$ 만이 필요 한데 이때 $m=1, 2, \dots$ 이며 창 크기를 결정한다. 학습 횟수가 증가함에 따라 정보 집합은 다만 $\{\Omega^*$ (aff. bfb. gs. $w_{\lambda\lambda}$)(k), $e^*(k)\}$ 로 감소하며 이것은 DNU 매개변수와 오차가 최적치에 수렴하였음을 가리킨다. 이를 위해 각 매개변수 벡터에 대해 최적화되어야 하는 성능 지표는 다음과 같이 정의되며 이때 E는 기대 연산자(expectation operator)이다.

$$J = E\{[e(k); \Omega$$
(aff. bfb. gs. $w_{\lambda\lambda})]\} \quad (14)$

식 (14)에서 $E\{e(k); \Omega$ (aff. bfb. gs. $w_{\lambda\lambda})\}$ 의 일반적인 형태는 오차의 우함수이다. 즉,

$$J = \frac{1}{2} E\{[e^2(k); \Omega$$
(aff. bfb. gs. $w_{\lambda\lambda})]\} \quad (15)$

이때 E는 기대 연산자이고 $e(k)$ 는 목표 신호 $u_d(k)$ 와 실제 신호 $u(k)$ 간의 차이로 정의되는 오차 신호이다. 벡터 Ω (aff. bfb. gs. $w_{\lambda\lambda}$)의 각 성분은 급경사 감소(steepest-descent) 알고리즘에 의해 J를 최소화 하는 방식으로 적용되었다. 급경사 감소법에서 매개변수 벡터의 조절은 성능지표 J의 음의 기울기에 비례하도록 되어진다. 즉,

$$\delta \Omega$$
(aff. bfb. gs. $w_{\lambda\lambda}$)(k) \propto $(-\nabla J)$ where,

$$\nabla J = \frac{\delta J}{\delta \Omega$$
(aff. bfb. gs. $w_{\lambda\lambda})} \quad (16)$

따라서, $\text{dia}[\mu]$ 가 독립적인 적용 이득 행렬일 때 다음과 같이 된다.

$$\delta \Omega$$
(aff. bfb. gs. $w_{\lambda\lambda}) = \\ -\text{dia}[\mu] \frac{\delta J}{\delta \Omega$ (aff. bfb. gs. $w_{\lambda\lambda})} \\ = -\text{dia}[\mu] \nabla J \quad (17)$

위의 식에서 $\text{dia}[\mu]$ 는

$$\text{dia}[\mu] = \begin{bmatrix} \mu_{ai} & 0 & 0 & 0 \\ 0 & \mu_{bj} & 0 & 0 \\ 0 & 0 & \mu_{gs} & 0 \\ 0 & 0 & 0 & \mu_{\lambda\lambda} \end{bmatrix} \quad (18)$$

이때 μ_{ai} , $i=0, 1, 2$, μ_{bj} , $j=1, 2$, μ_{gs} 는 DNU의 적응 매개변수의 독립적인 학습 이득이고, $\mu_{\lambda\lambda}$ 는 자기 및 상호 뉴런 시냅스 연결의 학습 이득을 나타낸다. ϕ (aff. bfb)로 DNU의 시냅스 가중치 벡터를 나타내면 ϕ (aff. bfb)에 대한 성능 지표의 기울기는 다음으로 구해진다.

$$\frac{\delta J}{\delta \phi$$
(aff. bfb)} = \frac{1}{2} E \left[\frac{\delta [u_d(k) - u(k)]^2}{\delta \phi(aff. bfb)} \right] \\ = E \left[e(k) \left\{ -\frac{\delta \Psi(v)}{\delta \phi(aff. bfb)} \frac{\delta v}{\delta \phi(aff. bfb)} \right\} \right] \\ = E \left[e(k) \{ \text{sech}^2[v(k)] P \phi(aff. bfb) \} \right] \quad (19)

이때

$$P \phi$$
(aff. bfb)(k) = $\frac{\delta v(k)}{\delta \phi$ (aff. bfb)} \\ = g_s \frac{\delta v_1(k)}{\delta \phi(aff. bfb)} \text{는 매개변수-상태(또는 감도)} \\ \text{신호의 벡터를 나타낸다. [9-10].} \\ P \phi(aff. bfb)(k) = $g_s [s(k-i)]$, $i=0, 1, 2$ $P \phi$ bfb(k) \\ = $-g_s [v_1(k-j)]$, $j=1, 2 \quad (20)$

비슷한 방법으로, somatic gain g_s 에 대한 성능지표의 기울기는 다음 식으로 주어진다.

$$\frac{\delta J}{\delta g_s} = \frac{1}{2} E \left[\frac{\delta [u_d(k) - u(k)]^2}{\delta g_s} \right] = E \left[-e(k) \{ \text{sech}^2[v(k)] v_1(k) \} \right] \quad (21)$$

자기 및 상호 뉴런 시냅스 연결에서의 적응은 다음과 같이 구할 수 있다.

$$\begin{aligned} \frac{\delta J}{\delta w_{\lambda\lambda'}} &= \frac{1}{2} E \left[\frac{\delta [u_d(k) - u(k)]^2}{\delta w_{\lambda\lambda'}} \right] \\ &= E \left[-e(k) \{ \text{sech}^2[v(k)] g_s u_\lambda(k-1) \} \right] \\ &= E \left[-e(k) \left\{ \frac{\delta \Psi(v)}{\delta v} \frac{\delta v}{d w_{\lambda\lambda'}} \right\} \right] \end{aligned} \quad (22)$$

위의 식들로부터 DNP 매개변수 갱신 알고리즘들은 다음과 같이 쓸 수 있다.

$$a_m(k+1) = a_m(k) + \mu a_i E[e(k) \text{sech}^2[v(k)] P \phi a_m(k)], i=1,2 \quad (23a)$$

$$b_{m_i}(k+1) = b_{m_i}(k) + \mu b_i E[e(k) \text{sech}^2[v(k)] P \phi b_{m_i}(k)], i=1,2 \quad (23b)$$

$$g_s(k+1) = g_s(k) + \mu g_s E[e(k) \text{sech}^2[v(k)] v_1(k)] \quad (23c)$$

$$w_{\lambda\lambda'}(k+1) = w_{\lambda\lambda'}(k) + \mu \lambda \lambda' E[-e(k) \text{sech}^2[v(k)] g_s u_\lambda(k-1)] \quad (23d)$$

3.2 비선형 동적 시스템 제어기법

비선형 플랜트의 제어상태를 최적으로 유지시키기 위해서는 DNP 제어기가 신경회로망의 초기포화 상태를 인식하고 비선형 활성화 함수의 기울기를 적절히 변화시켜주어야 한다. 기존의 DNU, DNP제어기의 비교를 위해 주어지는 파라미터 값은 동일한 값으로 하였다. 따라서 DNP의 초기 가중치는 0.8694로 고정하고, 학습을 $\mu=0.0003$, 그리고 운동량 항의 값은 0.15, 이산시간은 800으로 설정하였다. 이를 바탕으로 다음과 같은 3가지 방법의 시뮬레이션을 행하였다.[6-8]

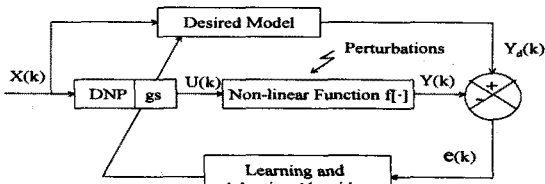


그림 3 DNP 알고리즘을 이용한 비선형 동적 시스템 제어기법

3.3 컴퓨터 모의 실험

경우1. 미지의 비선형 모델 제어하고자 하는 플랜트의 수학적 모델은 식 (24)와 같은 차분 방정식으로 기술된다.

$$y(k) = f[y(k-1), y(k-2), u(k), u(k-1), u(k-2)] \quad (24)$$

이때 미지의 비선형 함수 $f[\cdot]$ 은 식 (25)과 같다.

$$f[\cdot] = [2 + \cos\{7\pi(y^2(k-1) + y^2(k-2)) + e^{-u(k)}\} / 1 + u^2(k-1) + u^2(k-2)] \quad (25)$$

여기서 시스템 입력 $x(k) = \sin(2\pi k/250)$ 이다.

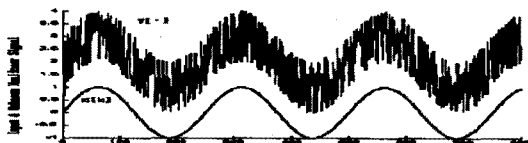


그림 4 경우 1의 시스템 입력 $x(k)$ 와 미지의 비선형 함수 $F[\cdot]$ 특성

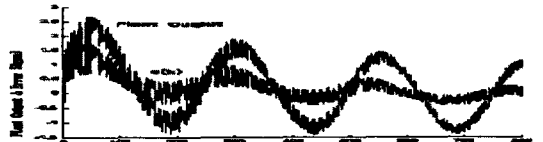


그림 5 DNU 제어의 100번째 학습에서의 플랜트 출력과 오차 응답 $e(k)$

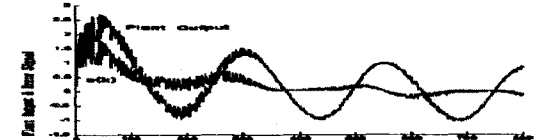


그림 6 DNP 제어의 100번째 학습에서의 플랜트 출력과 오차 응답 $e(k)$

경우2 미지의 비선형 특성이 변화하는 플랜트 제어 미지의 비선형 함수 $f[\cdot]$ 은 식 (26), (27) 그리고 식 (28) 같이 제어 공정이 진행되는 동안 세계의 비선형 함수로 변화한다.

$$f[\cdot] = \frac{[\sin\{\pi(y^2(k-2) + 0.5)\} + 0.3 \sin(2\pi u(k))]}{1 + u^2(k-1) + u^2(k-2)} \quad (26)$$

for $0 \leq k < 99$ and $500 \leq k < 599$

$$f[\cdot] = e^{-(y^2(k-1) + y^2(k-2))} + \sqrt{|u^2(k) + u^2(k-1) + u^2(k-2)|} \quad (27)$$

for $200 \leq 299, 400 \leq 499$ and $600 \leq k < 800$

$$f[\cdot] = \frac{[0.5 - 0.5 \cos y^2\{7\pi(y^2(k-1) + y^2(k-2))\}] + e^{-u(k)}}{4 + u^2(k-1) + u^2(k-2)} \quad (28)$$

for $100 \leq k \leq 199$ and $300 \leq k < 399$

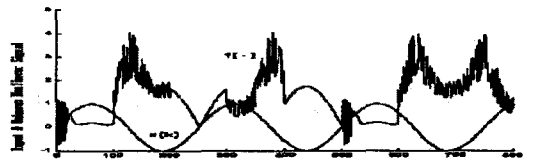


그림 7 경우 1의 시스템 입력 $x(k)$ 와 미지의 비선형 함수 $f[\cdot]$ 특성

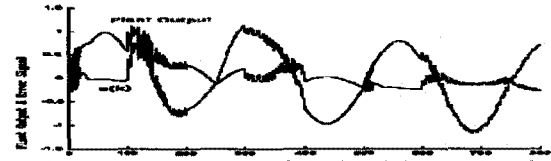


그림 8 DNU 제어의 100번째 학습에서의 플랜트 출력과 오차 응답 $e(k)$

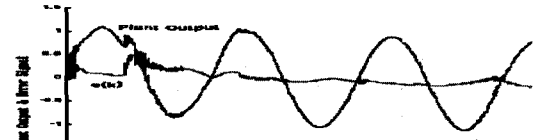


그림 9 DNP 제어의 100번째 학습에서의 플랜트 출력과 오차 응답 $e(k)$

경우3 미지의 비선형 특성이 변화하는 플랜트 제어 플랜트와 미지의 비선형 함수 $f[\cdot]$ 은 경우 1에서와 같이 동일하며 입력 신호 $x(k)$ 가 식 (29)과 같이 변화한다.

$$\begin{aligned}
 x(k) &= \cos(2\pi k/250) \quad 0 \leq k < 75 \\
 x(k) &= 0.8 \quad \text{for } 76 \leq k < 125 \\
 x(k) &= 0.6 \quad \text{for } 126 \leq k < 175 \\
 x(k) &= 0.4 \quad \text{for } 176 \leq k < 300 \\
 x(k) &= -0.6 \quad \text{for } 301 \leq k < 350 \\
 x(k) &= 0.5 \cos(2\pi k/150) \quad \text{for } 351 \leq k < 600 \\
 x(k) &= \cos(2\pi k/250) \quad \text{for } k \geq 601
 \end{aligned}
 \tag{29}$$

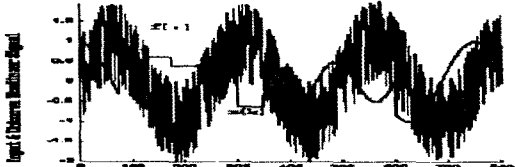


그림 10 경우 3의 시스템 입력 $x(k)$ 와 미지의 비선형 함수 $f(\cdot)$ 특성

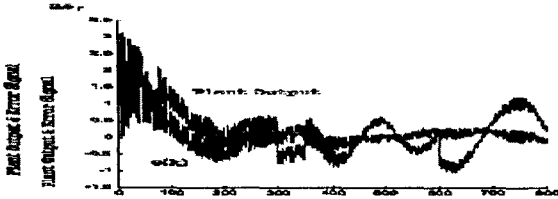


그림 11 DNU 제어의 100번째 학습에서의 플랜트 출력과 오차 응답 $e(k)$

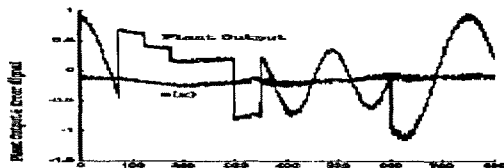


그림 12 DNP 제어의 50번째 학습에서의 플랜트 출력과 오차 응답 $e(k)$

4. 결 론

DNP 제어기는 플랜트의 기준 입력의 변화와 외란의 영향, 시스템 파라미터값의 변화 등 제어환경의 변화에 대하여 DNU 단독 제어기보다 추종성과 강인성, 적응성 등에 대하여 전체적 수렴속도가 향상됨을 알 수 있었다. 즉, DNU 단독의 학습 알고리즘에 의한 비선형 동적 시스템은 100번째 학습후에 적응 추종함을 보이지만, 본 논문에서 제안한 DNP 논리 알고리즘에 의한 시스템 제어는 50번째 학습부터 적응 추종이 이루어짐을 알 수 있어 전자보다 빠른 학습수렴과 제어성능이 향상됨을 보여 주었다. 이는 플랜트 출력이 입력 신호에 잘 적응함을 보여준다.

(참 고 문 헌)

- [1] K.s Narendra and K. Parthasarthy, "Identification and Control of Dynamical System Using Neural Network." IEEE Trans. Neural Network, pp.1-14, 1990.
- [2] 조현섭, "A Simple and Effient Technique for Rapid Convergence Speed of The LMS Algorithm", ITC-CSCC'97(Japan,Okinawa),(Vol 1),pp31-34,199.7
- [3] Madan M. Gupat, "Fuzzy Logic and Neural Networks", Proceeding of the Ten International Conference(TAIPEI'92), Vol 3, pp281-294,July,1992
- [4] Robert Hecht-Nielsen, "Neural Computing", HNC, Inc. and University of California San Diego, 1991
- [5] D. Stipanicev, M. De Neyer, and R. Gorez,

- "Self-tuning Self-organizing Fuzzy Robot Control", Proc., IFAC Symp. Robot Control SYROCO'91, Vienna, Sept권., 1991
- [6] M. M. Gupta and Rao, "Neural Learning of Robot Inverse Kinematics Transformations", In Neural and Fuzzy System: The Emerging Science of Intelligent Computing, S. Mitra, W. kraske, and M. M. Ghpta, Eds, SPIE Institute Series Editor, Bellingham, Washington, 1995
 - [7] S. W. Park and B. H. 대, "Design of Nonlinear System using Dynamic Neural Networks.", proc., KACC, pp60-64, March 1995
 - [8] L. H. Tsoukalas and R. E. Uhrig, "Fuzzy and Neural Approaces in Engineering", Jone Wiley & Sons, Ins., 1996