

다층 신경회로망을 이용한 유연성 로봇트팔의 위치 제어

(Position Control of a One-Link Flexible Arm Using Multi-Layer Neural Network)

金秉燮*, 沈貴寶**, 李鴻奇**, 全洪兌*

(Byung Seob Kim, Kwee Bo Sim, Hong Gi Lee, and Hong Tae Jeon)

要 約

본 논문에서는 단일 링크 유연성 로봇트 팔의 위치 제어를 위해 신경망 제어기를 제안한다. 기본적으로 제어기는 다층 신경 회로망과 PD 제어기가 병렬로 구성되어 있다. 신경회로망은 오차 역전파 학습 법칙에 의해 학습이 이루어 지며, 학습하는 동안, 신경회로망의 가중치들은 원하는 허브각과 실제 허브각 사이의 위치오차를 최소화하도록 조정된다. 끝으로 제안된 접근방식의 효율성은 컴퓨터 시뮬레이션에 의해 증명된다.

Abstract

This paper proposes a neuro-controller for position control of one-link flexible robot arm. Basically the controller consists of a multi-layer neural network and a conventional PD controller. Two controller are parallely connected. Neural network is trained by the conventional error back propagation learning rules. During learning period, the weights of neural network are adjusted to minimize the position error between the desired hub angle and the actual one. Finally the effectiveness of the proposed approach will be demonstrated by computer simulation.

I. 서 론

대부분의 로봇트 매니플레이터는 강체(rigid body)로 이루어져 있어 작동시 큰 구동력이 필요하며 많은 에너지가 소모되고, 또한 섬세한 힘이 요구되는 작업(예를 들면 유리창과 같은 약한 표면을 일정한

힘을 주면서 궤적을 추종하는 작업)에는 제어하는데 어려움이 존재한다.

이러한 단점을 보완하기 위해 질량이 가벼운 유연성 매니플레이터에 대한 연구가 현재 활발히 진행되고 있다. 유연성 매니플레이터는 질량이 가벼워 적은 에너지 소모, 및 빠른 구동등의 잇점을 갖고 있으며, 섬세한 힘을 요구하는 작업에는 강건체의 매니플레이터 보다 더 효율적이다. 그러나 유연성 로봇트는 구동시 팔의 힘과 진동으로 인해 기존의 로봇트 보다 그 위치 제어가 더 어렵다.

Cannon, Schmitz^[1]는 유연성 로봇트의 tip position 제어를 위한 제어시스템을 제안하였다. 이 방식에서는 tip position의 진동을 억제하기 위해 시스템의 특성방정식의 pole과 zero들을 안정화시키는 estimator

*正會員, 中央大學校 電子工學科
(Dept. of Elec. Eng., Chungang Univ.)

**正會員, 中央大學校 制御計測工學科
(Dept. of Cont. Inst. Eng., Chungang Univ.)

接受日字: 1991年 8月 9日

(※ 본 연구는 한국과학재단의 목적 기초 연구비 지원으로 이루어 졌음.)

를 제어기에 부착하여 실험했으나, 센서와 기타 보조 장비가 많이 필요하고 제어기가 복잡한 단점을 갖는다. 한편, Wang, Lu, Hsu^[2]는 간단한 PD(propotional dervative) 제어기로써 허브 위치각을 제어하는 방식을 제안하였다. 이 방식은 비교적 간단한 제어방식이기는 하나 섬세하고 정확한 제어가 요구될 때 그 효율성을 잃게 된다. 그리고 Bayo, Moulin^{[3][4]}는 FEM 방식을 이용하여 유연성 로봇 팔의 역동력학을 풀이하였으며 복잡한 식을 간단히 하기 위해 direct-integration 방법을 도입하였다. 그러나 이 방식은 동적 파라미터들의 정확한 추정을 필요로 하는 문제점을 갖는다.

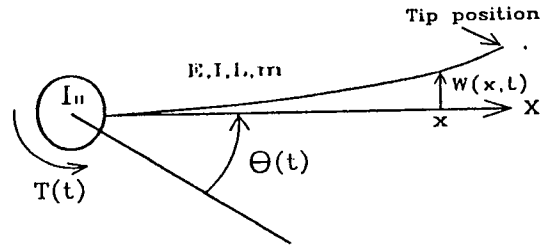


그림 1. 단일 링크 유연성 로봇 팔
Fig. 1. A single-link flexible robot arm.

이에 본 논문에서는 복잡한 비선형 시스템의 정확한 동적 모델링이 필요없고 외부 및 내부의 동적 조건의 변화에도 자기 조정능력을 갖는 다층 신경회로망 기법을 도입하여 유연성 매니퓰레이터의 제어기를 구성한다. 제안하는 유연성 매니퓰레이터의 제어기는 기존의 PD 제어기에 신경 회로망을 병렬로 연결하여 학습 초기에는 PD 제어기가 주된 작용을 하며 학습후에는 신경 회로망이 주된 작용을 하는 전체 구조를 가진다. 이러한 신경 제어기의 장점은 다음과 같다.

2. $\dot{\theta}^2$ 와 관련된 모든 항은 무시한다.
3. 회전 inertia와 shear deformation 효과를 무시하는 Euler-Bernoulli 빔 모델을 사용한다.
4. 중력을 무시한다. 즉 그림(1)의 x, y 평면이 지상과 평행하다.

상기 가정하에 허브의 관성 모멘트를 I_h , 빔의 휨으로 인한 deflection을 w , 허브가 회전한 각도를 $\theta(t)$ 라 정의할 때 임의의 x 점에서 시간 t 초후에 end-effector가 간 거리를 $y(x, t)$ 라 하면,

$$y(x, t) = w(x, t) + x\theta(t) \tag{1}$$

로 표현할 수 있다.

이때 운동에너지 T_k 와 위치에너지 V 를 정의하고 Hamilton의 법칙과 Lagrangian 방식을 이용하여 다음과 같은 무한대의 상미분 방정식을 얻을 수 있다.^[1]

$$\frac{dx}{dt} = Fx + GT(t) \tag{2}$$

$$z = Hx \tag{3}$$

여기서, $F \in \mathbb{R}^{(2n+2) \times (2n+2)}$, $x \in \mathbb{R}^{2n}$, $G \in \mathbb{R}^{2n}$, $H \in \mathbb{R}^{2 \times 2n}$, $z \in \mathbb{R}^2$ 는 아래와 같은 행렬이다.

$$x = [q_0 \ \dot{q}_0 \ q_1 \ \dot{q}_1 \ \dots \ q_n \ \dot{q}_n]^T \tag{4}$$

$$F = \begin{bmatrix} 0 & 1 & & & & \\ 0 & 0 & & & & \\ & & 0 & 1 & & 0 \\ & & -\omega_1^2 & -2\zeta_1\omega_1 & & \\ & & & 0 & 0 & 1 \\ & & & & -\omega_n^2 & -2\zeta_n\omega_n \end{bmatrix} \tag{5}$$

$$G = \frac{1}{I_r} \begin{bmatrix} 0 & 1 & 0 & \frac{d\phi_1}{dx}(0) & \dots & 0 & \frac{d\phi_n}{dx}(0) \end{bmatrix}^T \tag{6}$$

II. 유연성 로봇의 동적 모델링과 그 문제점

유연성 매니퓰레이터의 동적 모델을 얻기 위한 방법은 크게 유한요소법(Finite Element Method)^{[3][4]}과 modal analysis에 의한 탄성변형의 근사화를 근거로한 Lagrangian방식^[1]등 두가지로 나눌 수 있다. 본 절에서는 Lagrangian 방식에 의한 단일 링크 유연성 로봇의 동적 모델링 방식을 간략하게 살펴본다.

일반적으로 Lagrangian 방식에 의한 동적 모델링은 다음과 같은 가정들을 갖는다.

1. w (deflection)은 0.1L이하이다.

$$H = \begin{bmatrix} L & 0 & \phi_1(L) & 0 & \dots & \phi_n(L) & 0 \\ 1 & 0 & \frac{d\phi_1}{dx}(0) & 0 & \dots & \frac{d\phi_n}{dx}(0) & 0 \end{bmatrix} \quad (7)$$

$$z = [y(L, t) \ \theta_n(t)]^T \quad (8)$$

여기서, $n=0$ 일 때 강체모드를 나타내어 $\phi_0(x) = x, q_0(t) = \theta(t)$ 가 된다. $y(L, t)$ 는 길이 L 인 유연성 로봇 팔끝이 초기 위치에서 시간 t 초후에 간 거리이며, $\theta_n(t)$ 는 시간 t 초후에 회전한 허브각(hub angle) 이다.

그러나 상기식을 이용하여 제어기를 구성할 경우 다음과 같은 어려운 문제점들을 갖는다.

1) 실제 제어 시스템 구현시 식(4)의 상태벡터 x 값을 구하기 위한 ξ , 및 ω 등은 정밀한 계측장비에 의해 측정되어 구해지는 값들이다. 측정 오차가 존재하고 오차가 누적되어 효율적인 제어기 구성이 용이하지 않다.

2) 유연성 로봇트는 외부 환경과 부하의 변화에 민감하고, 시간의 경과로 인한 마모가 심하므로 파라미터의 변화가 비교적 많다.

상기 어려움들을 효율적으로 극복할 수 있는 방법 중의 하나는 학습 및 병렬계산구조를 갖는 신경회로망 기법을 응용하여 유연성 매니퓰레이터의 신경제어기를 구성하는 일이다.

III. 유연성 로봇트의 신경 위치 제어기

1. 다층 신경회로망

다층 신경 회로망(multi-layer neural network)을 구성하는 각 인공뉴런(artificial neuron)은 다음과 같은 활성화 함수(activation function)를 갖는다.

$$f(X) = 1 / (1 + \exp(-aX)) \quad (9)$$

여기에서 $a(0 \leq a \leq 1)$ 는 sigmoid 함수의 기울기를 정하는 계수값이다.

식(11)의 sigmoid 함수를 이용 각 뉴런의 입-출력 특성은 다음과 같이 표현할 수 있다.

$$NET = x_1 w_1 + x_2 w_2 + \dots + x_n w_n = \sum_{i=1}^n x_i w_i \quad (10a)$$

$$OUT = f(NET) \quad (10b)$$

여기서, NET는 뉴런의 입력 x_1, x_2, \dots, x_n 와 입력에 해당하는 가중치 w_1, w_2, \dots, w_n 의 곱으로 결정되는 값이며, OUT는 NET 값이 활성화함수를 통과하여 최종적으로 각 뉴런들의 출력값이 된다(그림2 참조).

상기 인공 뉴런들을 여러층으로 나열하고 각 뉴런들을 상호 연결하면 다층 신경회로망을 구성할 수 있다(그림4 참조).

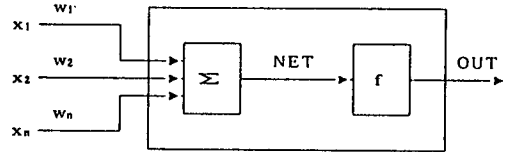


그림 2. 인공 뉴런
Fig. 2. Artificial neuron.

2. 유연성 로봇트의 신경 제어기

유연성 매니퓰레이터의 제어 대상으로는 tip position과 허브각으로 크게 나눌 수 있다. 유연성 매니퓰레이터의 특성상 tip position의 제어는 중요하나 센싱을 위한 복잡한 주변기기의 필요성 그리고 복잡한 제어기법으로 인해 어려움이 존재한다. 이에 본 논문에서는 허브각을 센싱한다고 가정하여 제어기를 구성하였다. 이에 대한 장점은 센싱하기가 용이하여 복잡한 주변기기의 필요성이 없게되며 허브각의 궤적(trajjectory)를 적절히 정해주면 tip position 역시 진동없이 결과값에 도달할 수 있다는 점이다.²⁾

본 논문에서 제안하는 유연성 로봇트의 신경 제어기는 그림3과 같다. 그림3에서 다층 신경망으로 구성된 신경제어기와 기존의 PD 제어기가 병렬로 연결되어 있다. 본 제어기를 설명하기 위해 전방향(feedforward) 제어, 역방향(feedback) 제어 그리고 전체적인 제어과정으로 나누어 살펴 보겠다.

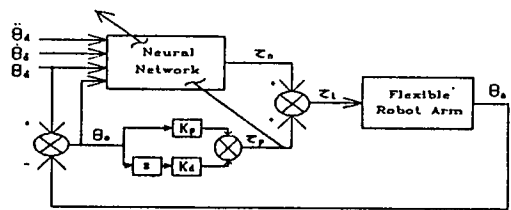


그림 3. 유연성 로봇트의 신경제어기
Fig. 3. The neuro-controller of the flexible robot arm.

1) 전방향(feedforward) 제어

허브 각도(θ_a), 허브 각속도($\dot{\theta}_a$), 허브 각가속도($\ddot{\theta}_a$)의 명령 궤적이 제어기의 입력으로 주어진다. 이 세개의 입력은 신경 회로망으로 인가되고 이 중에서 θ_a 는 출력 각도(θ_a)와 비교되어 식(13)에 의해 그 차(θ_e)가 PD제어기 입력과 신경 회로망의 네번째 입력으로 들어간다.

$$\theta_e = \theta_a - \theta_s \quad (11)$$

θ_e 가 네번째 입력으로 들어가는 이유는 다음과 같다. 강체 로봇트 팔과는 달리 유연성 로봇트 팔의 경우 원하는 최종 결과에 도달하였을 때에도 진동없이 그 값을 유지해야 하므로 비록 작기는 하지만 토오크가 존재해야 할 필요성이 있다. 그러나 최종 결과 시간 이후에 원하는 궤적들의 값은 모두 동일하다. 즉 θ_a 와 θ_s 는 zero이며, $\dot{\theta}_a$ 는 원하는 각도값에 고정되어 있다. 그런데도 최종 결과 시간 이후에도 학습이 이루어져 신경회로망의 토오크가 존재한다. 그러기 위해서 새로운 입력이 추가되어야 할 필요성이 있다. θ_a 를 입력으로 인가할 수 있으나 이 값은 거의 θ_s 와 같기 때문에 별 의미가 없음을 알 수 있다. 그래서 θ_e 를 새로운 입력으로 인가한다. 비록 작은 값이지만 부호의 변화에 따라 그 해당하는 가중치 값들이 크므로 출력에 영향을 줄 수 있기 때문에 신경 회로망의 최종 결과시간 이후의 학습을 담당할 수 있다.

이 네 입력에 대해 신경회로망은 신경망 토오크 (τ_n)를 출력한다. PD 제어기는 θ_e 와 $\dot{\theta}_e$ 에 의해 PD 토오크 (τ_p)를 출력한다.

$$\tau_p = K_p \theta_e + K_d \dot{\theta}_e \quad (12)$$

여기서 K_p 와 K_d 는 각각 위치오차와 속도오차의 이득 상수들이다. 결국 두 토오크가 더해져서 유연성 로봇트 팔의 입력으로 인가된다. 즉,

$$\tau_t = \tau_p + \tau_n \quad (13)$$

유연성 로봇트 팔은 입력된 값만큼 회전하며, 센서를 이용하여 실제 허브각(θ_s)을 얻을 수 있다.

2) 역방향(feedback) 제어

얻어진 θ_s 는 입력쪽으로 feedback되어 θ_a 와 비교된다. 한편, 신경회로망은 출력쪽의 에러가 가중치를 변하게 하고 이 가중치 변화가 다시 hidden Layer의 가중치를 변하게 하여 결국 에러가 출력에서 입력쪽으로 전달되는 역전파 방식에 의해 학습이 이루어진다. 본 논문에서는 학습시간을 보다 빠르게 할 수 있는 방법으로 알려진 momentum 방식을 적용한다.^{[8][9]} 출력층의 경우 가중치 변화율과 변화된 가중치는 다음과 같다.

$$\delta_{r,i} = error_{r,i} \cdot f'(NET_{r,i}) \quad (14a)$$

$$error_{r,i} = target_{r,i} - OUT_{r,i} \quad (14b)$$

$$\Delta W_{q,r,i}(n+1) = \eta \cdot \delta_{r,i} \cdot OUT_{q,i} + \alpha \cdot \Delta W_{q,r,i}(n) \quad (14c)$$

$$W_{q,r,i}(n+1) = W_{q,r,i}(n) + \Delta W_{q,r,i}(n+1) \quad (14d)$$

α = Momentum 상수

η = 학습율 (learning rate)

$W_{p,r,i}(n)$ = Hidden 층의 뉴런 q로부터 출력층은 뉴런 r을 연결해 주는 가중치 값. (첨자1는 이 가중치의 목표가 되는 층을 나타낸다.)

$W_{p,r,i}(n+1)$ = step n+1(조정후)에서의 가중치값.

$\delta_{r,i}$ = 출력층의 뉴런 r에 해당하는 δ 값.

$OUT_{q,k}$ = Hidden층의 뉴런 p에 해당하는 OUT값.

$NET_{r,i}$ = 출력층의 뉴런 r에 해당하는 NET값.

그림4와 식(14)의 가중치를 나타내기 위한 첨자 중에서 p, q, r는 해당 layer의 특정 뉴런이고 j, k, l는 layer를 표현한다.

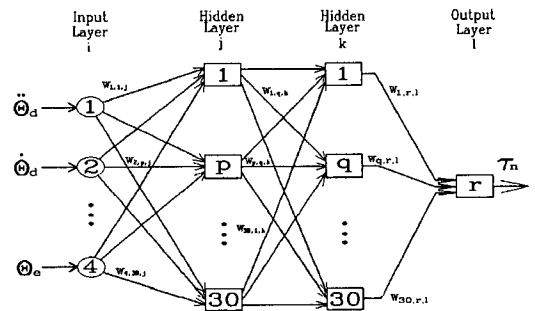


그림 4. 제어기에 이용된 3층 신경회로망
Fig. 4. 3-layer neural network used in the controller.

$\delta_{r,i}$ 를 구하기 위해서는 목표(target) 값을 알아야 하지만 목표값을 구할 수 없으므로 적절한 error 값에 해당하는 값을 구해야 한다. 이에 신경회로망의 에너지함수를 다음과 같이 정의한다.

$$E = 1/2 \tau_p^2 \quad (15)$$

식(15)은 오차가 제로이면 에너지도 제로가 되므로 적절한 에너지 함수라 할 수 있다. 이 식(15)로부터 출력층의 $\delta_{r,i}$ 값은 다음과 같이 구할 수 있다.^{[7][11]}

$$\delta_{r,i} = \tau_p \cdot f'(NET_{r,i}) \quad (16)$$

그러므로, 식(14b)로부터

$$error_{r,i} = \tau_p \quad (17)$$

가 되어 error값으로 τ_p 가 신경회로망으로 인가된다.

Hidden layer의 경우 $\delta_{p,i}$ 값은 바로 앞층에서 구한 $\delta_{q,k}$ 를 구하고자 하는 뉴런q와 k층의 모든 뉴런에 연

결된 가중치와 곱한 값을 모두 더하여 해당 뉴런 q의 입력에 대한 활성화 함수의 미분한 값을 곱해서 구할 수 있다. 식으로 표현해 보면,

$$\delta_{p,q} = \left(\sum_k \delta_{q,k} \cdot w_{p,q,k} \right) \cdot f' (NET_{p,q}) \quad (18)$$

이 되고 가중치 변화율과 변화된 가중치는 위의 식(14c), 식(14d)와 같은 형태이다. 식(14)~식(18)의 과정을 거쳐서 error가 출력에서 입력으로 역전파되면서 학습을 이루어 나간다.

3) 전체적인 제어 과정

신경회로망의 error 값으로 τ_p 가 인가되기 때문에 τ_p 를 최소화하는 방향으로 학습한다. 학습 과정에서 신경회로망의 가중치들은 유연성 로봇의 역동력과 일치하는 방향으로 조정되어 지며 수렴하게 된다. 그러므로 학습초기의 τ_n 는 주로 τ_p 로 이루어져 있지만 학습이 이루어져 갈수록 θ_e 는 작아지기 때문에 τ_p 가 작아지고 τ_n 이 τ_t 를 대부분 차지하게 된다. 학습이 만족할 만큼 이루어지면 신경 회로망의 출력만으로 제어할 수 있게 된다. 위 과정에서 알 수 있는 바와 같이 시스템의 파라미터나 제어에 필요한 복잡

한 식의 요구없이 스스로의 학습으로 제어가 이루어지는 것이다. 또한 신경 제어기는 PD제어기로 정확히 추종할 수 없는 궤적을 추종할 수 있으며, 로봇 팔의 마모나 기타 다른 이유로 인한 파라미터의 변화에 효과적으로 대처할 수 있다. 본 논문의 전체 학습 순서도를 다음과 같이 요약할 수 있다.

IV. 모의실험 및 결과

1. Simulator

본 논문에서 이용한 유연성 매니플레이터의 simulator는 컴퓨터 시뮬레이션으로 식(4)~(10)과 같은 상태 방정식을 풀어 허브각과 간거리를 결과로 얻는 방법을 이용하여 만들었으며, 상태 방정식의 풀이는 비교적 정밀한 Runge-Kutta 수치해석법을 이용하여 풀이하였고, 상태 방정식에 필요한 파라미터들은 제어기의 일반성을 위해 Schmits, Cannon의 내용을 그대로 이용하였다(표1 참조).

표 1. 처음 3개에 플렉서블 모델을 위한 매개변수

Table 1. Model parameter for first three flexible modes.

| Mode Number | Pinned-free Frequency F_1 (Hz) | Modal Damping ζ_1 | Actuator Modal Gain $d\phi/dx(0)$ (rad) | Tip-sensor Modal Gain $\phi_1(L)$ (m) |
|-------------|----------------------------------|-------------------------|---|---------------------------------------|
| 0 | 0 | 0 | 1.0 | 1.12 |
| 1 | 1.88 | 0.015 | 2.97 | -1.1 |
| 2 | 3.44 | 0.02 | 3.0 | 0.90 |
| 3 | 7.70 | 0.02 | 1.25 | -1.21 |

주어진 파라미터에 대한 flexible 로봇의 특성을 알아보기 위해 토크(10Nm)를 $t=0$ 초에서 $t=1$ 초까지 계속 인가하고 그 이후는 가하지 않았을 때의 허브각 응답을 구해보았다(그림6 참조). 그림에서 보는 바와 같이 진동이 무척 심한 것을 알 수 있다.

2. Trajectory

본 논문의 입력 궤적은 그림7과 같이 초기 각도(0°)에서 시작하여 원하는 각도(20°)까지 도달하도록 주어진다. Tip position의 진동을 최소로 하기 위해 각 가속도는 사인커브로 주어졌다. 각 속도 및 각도의 값들은 각가속도의 적분값들로 주어지며, 이 각각의 값들은 일정한 간격으로 샘플링되어 입력값으로 주어진다. 학습은 한 trajectory 구간 전체를 목표로 했으며 전 구간을 원하는 출력값이 나올때까지 반복하였다.

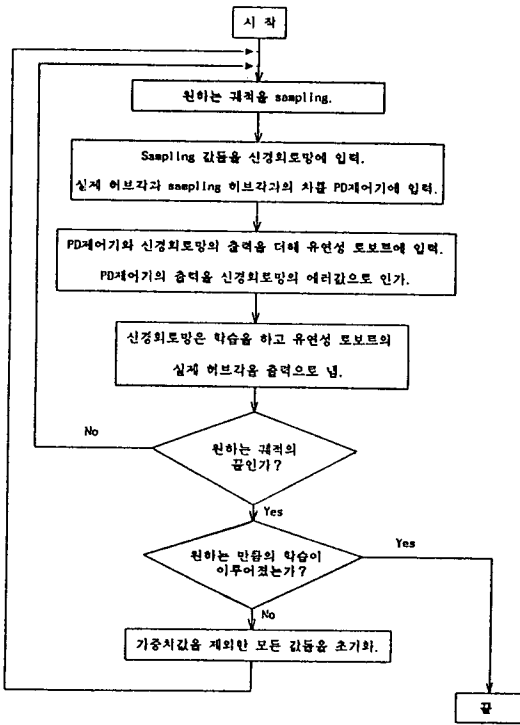


그림 5. 제안한 신경 제어기의 순서도
Fig. 5. Flow chart of the proposed neuro-controller.

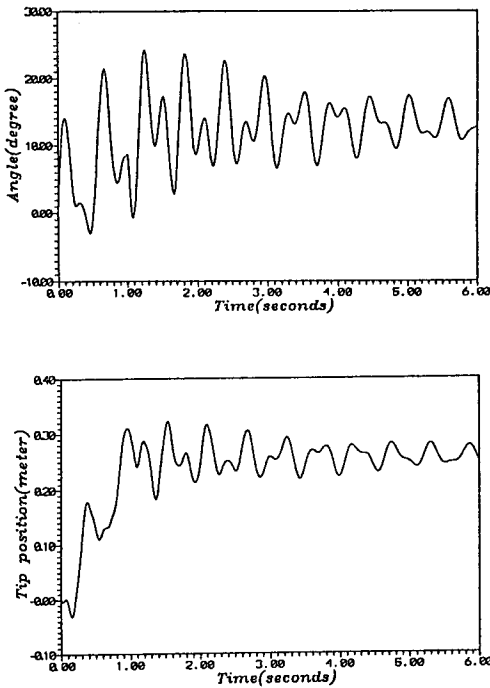


그림 6. 일정한 토크를 $t=0$ 에서 $t=1$ 초 사이에 가했을 때 유연성 로봇 팔의 위치
 Fig. 6. Position of flexible robot arm when constant torque is applied between $t=0$ and $t=1$.

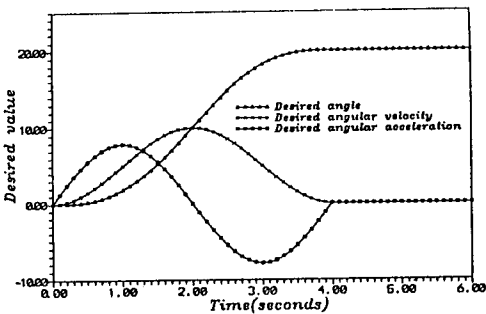


그림 7. 입력되는 궤적
 Fig. 7. The input trajectory.

V. 결 과

유연성 로봇 팔의 길이(L)는 1.12meter이고 허브에 영향을 미치는 총 관성모멘트(I_T)는 0.44kgm^2 이다. 컴퓨터 시뮬레이션시 샘플링 시간은 0.01초로 정했고 PD제어기의 이득들은 overshoot가 최소이면서 settling 시간이 빠른 $K_p=150.0$, $K_d=2.0$ 으로 잡

았다. 모든 입력되는 각도들은 radian값이다. 한편, 신경회로망에 쓰인 파라미터는 다음과 같다.

Hidden층은 둘이며 각 층에 30개의 뉴런을 연결하였다. 그러므로 총 가중치수는 $1050(4 \times 30 + 30 \times 30 + 30 \times 1)$ 개가 된다. η (학습율) 값은 0.2 α (momentum constant)는 0.1, a(activation 함수의 기울기) 값은 0.5이다. 신경회로망 출력 뉴런의 값은 범위를 알 수 없으므로 다른 뉴런들과 똑같은 활성화 함수를 사용한다면 원하는 결과값을 얻을 수가 없다. 그래서 출력 뉴런의 활성화 함수로 기울기가 0.02인 ramp 함수를 이용했다. 초기 가중치값들은 범위가 $-10 \sim +10$ 까지의 random수로 정해지게 된다.

학습초기에 실제각과 원하는 각과의 차이가 심하게 나타나 있는 것을 볼 수 있다(그림8 참조). 하지만 1000번 학습후 실제각과 원하는 각의 차는 거의 찾아볼 수 없다(그림9 참조). 학습초기에도 진동이 없는 것을 볼 수 있다. 그 이유는 PD 제어기만으로

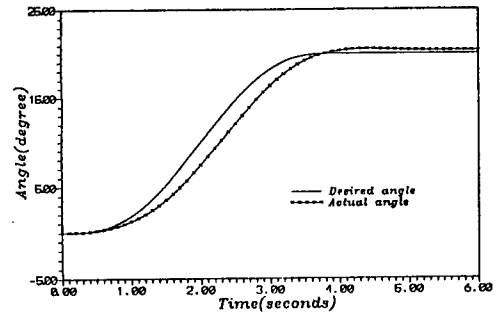


그림 8. 처음 학습시 원하는 궤적과 실제 궤적
 Fig. 8. The desired trajectory and actual one at the 1st learning trial.

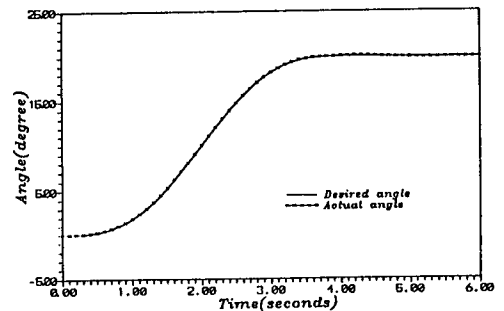


그림 9. 1000번 학습시 시간에 따른 원하는 궤적과 실제 궤적
 Fig. 9. The desired trajectory and actual one at the 1000th learning trial.

제어할 경우에도 적절한 K_p, K_d 를 구할 경우 진동없이 결과값에 도달하게 된다.²⁾ 하지만 원하는 궤적을 만족스럽게 추종할 수 없으며, 결과값에 도달하는 과정을 향상시킬 수 없다. 학습초기와 학습후 tip position의 위치 또한 살펴보았다(그림10 참조). 그림에서 보는 바와 같이 학습초기보다 학습이후 더욱 훌륭하게 결과값에 도달한다. 그림11과 그림12는 학습에 따른 PD 제어기의 출력과 신경회로망의 출력을 나타낸다. 학습 후 PD 제어기의 출력은 거의 제로 값에 가까운 것을 볼 수 있으며 신경 회로망의 출력은 학습초기에는 거의 제로인 값을 가지지만 학습 후에는 입력되는 토오르크를 대부분 담당하는 것을 볼 수 있다. 1000번 학습 후 학습된 가중치를 그대로 이용하여 tip position에 부하를 지닌 경우에 대해서도 모의 실험을 해보았다. 부하의 질량은 정확히 기재할 수 없는데 그 이유는 수학적 모델을 만들때 필요한 파라미터들은 만들어진 유연성 로봇 팔에 특별한 센서를 달아 측정할 값이므로 임의의 부하를 걸었을 때의 값을 추정할 수 없다. 부하가 걸려있을

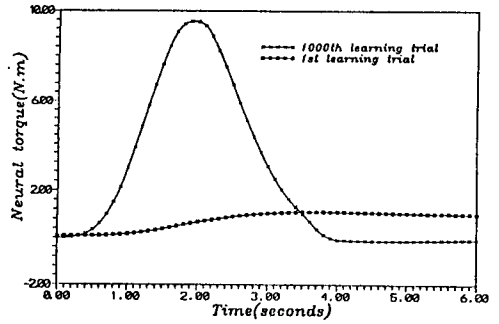


그림12. 학습에 따른 신경회로망의 출력
Fig. 12. Output of neural network at learning trial.

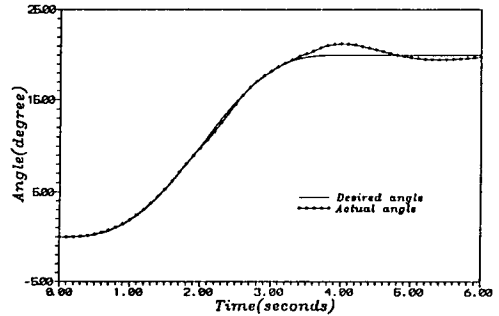


그림13. 1000번 학습후 부하를 달았을 때 원하는 궤적과 실제궤적 (F_1 30% 감소)
Fig. 13. Desired trajectory and actual one when load applied after 1000th learning trial (F_1 30% reduction).

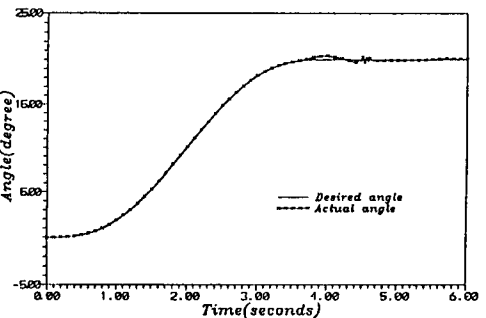


그림14. 부하를 단후 500번 학습시 원하는 궤적과 실제궤적 (F_1 30% 감소)
Fig. 14. Desired trajectory and actual one at 2500th learning trial after load applied (F_1 30% reduction).

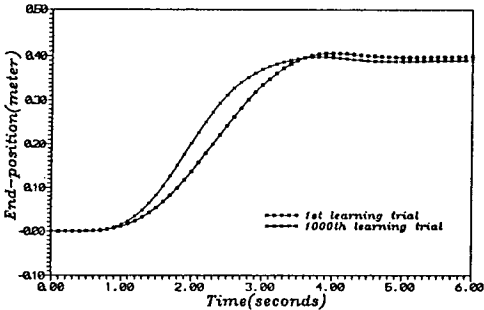


그림10. 처음 그리고 1000번 학습에서의 Tip position
Fig. 10. The tip position at the 1st and 1000th learning trials.

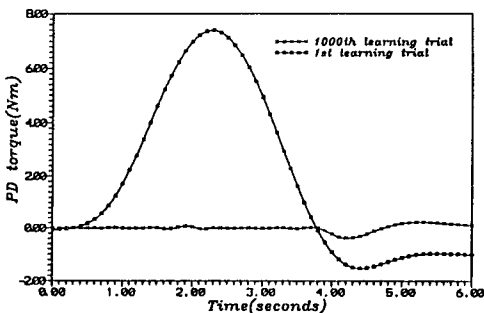


그림11. 학습에 따른 PD 제어기의 출력
Fig. 11. Output of DP controller at learning trial.

경우 pinned-free frequency (F_1)는 감소하므로 그 값들을 일정한 비율(30%)로 감소시켜 모의실험을 했다. 비록 파라미터의 급격한 변화가 있었지만 비교적 훌륭한 결과를 얻을 수 있었다(그림13, 14 참조). 부하를 건 후 바로 동작시켰을 경우 초기에는 훌륭히 추종하다가 정지시간 이후는 학습이 완벽히 이루어지지 못한 관계로 인해 신경회로망의 출력이 원하는 만큼의 값을 갖지 못하여 오차가 많이 발생하였다(그림13 참조). 500번 학습이후 오차는 많이 줄었으며 정지시간 이후(4초에서 5초사이) 약간의 오차가 존재하는 것을 볼 수 있는데 이것은 신경회로망의 완벽한 학습이 아직 이루어지지 못한 관계로 PD 제어가 보상하여 전체적인 균형을 이루어 나가는 과정이기 때문이다(그림14 참조).

V. 결론 및 앞으로의 연구방향

최근까지 유연성 매니플레이터에 대한 연구 결과들은 그 구조적 특성에 따른 동적 특성의 복잡성으로 인해 효율적인 제어 방식이 제안되지 못했다. 이에 본 논문에서는 구조적인 면에서 빠른 속도에서의 적용이 가능한 신경회로망을 이용하여 동적방정식이 필요치 않은 제어방식을 제안하였다. 학습이 진행되면서 신경 회로망이 주제어기로 동작하여 유연성 매니플레이터와 같은 고도의 비선형 시스템의 정보나 매개 변수의 지식이 없어도 제어가 가능함을 입증하였다. 그리고 학습이 이루어진 상태에서 부하를 첨가하였을 경우도 주어진 궤적을 잘 수행하였다.

앞으로는 실제 신경 회로망의 chip이 만들어져 실제 실험도 행해져야 할 것이며, 역전파 학습 알고리즘의 단점이라고 할 수 있는 학습시간을 줄일 수 있는 연구 또한 이루어져야 한다. 또한 one-link에서 two-link로의 확장에 관한 연구가 행해져야 하겠다.

參 考 文 獻

[1] R.H. Cannon, and E. Schmitz, "Initial Experiments on the End-point Control of a Flexible One-Link Robot," International

Journal of Robotics Research, vol. 3, no. 3, pp. 62-75, Fall 1984.

- [2] W.J. Wang, S.S. Lu, and C.F. Hsu, "Experiments on the Position Control of a One-Link Flexible Robot Arm," Trans. on Robotics and Automation, vol. 5, no. 3, June 1989.
- [3] E. Bayo, "A Finite-Element Approach to Control the End-Point Motion of a Single-Link Flexible Robot," *J. Robotic Systems*, vol. 4, no. 1, pp. 63-75, 1987.
- [4] E. Bayo, and H. Moulin, "An Efficient Computation of the Inverse Dynamics of Flexible Manipulators in the Time Domain," *Proceedings of 1989 IEEE International conference on Robotics and Automation*, pp. 710-715, 1989.
- [5] K.S. Rattan, V. Feliu, and H.B. Brown Jr., "Tip Position Control of Flexible Arms Using A Control Law Partitioning Scheme," *Proceedings of 1990 IEEE International Conference on Robotics and Automation*, pp. 1803-1808, 1990.
- [6] L. Meirovitch, *Elements of Vibration Analysis*. McGraw-Hill Co., N.Y., 1986.
- [7] M. Kawato, Y. Uno, M. Isobe, and R. Suzuki, "A hierarchical neural networks model for voluntary movement with application to robotics," *IEEE Control System Magazine*, vol. 8, pp. 8-16, 1988.
- [8] P.D. Wasserman, *Neural Computing, Theory and Practice*. Van Nostrand Reinhold, 1989.
- [9] C. Klimasauskas, J. Guiver, and G. Pelton, *Neuralworks Professional II*. Neural Ware, Inc., 1989.
- [10] 김기환, 강경운, 전홍태, "Timoshenko beam 모델을 이용한 두 개의 링크를 갖는 유연성 매니플레이터의 위치제어," '90 한국자동제어 학술회의 논문집, Vol. 1, pp. 382-387, 1990.
- [11] 이효중, 최영길, 전홍태, "신경 회로망을 이용한 단일 링크의 유연한 매니플레이터의 위치제어," '90 한국자동제어 학술회의 논문집, Vol. 1, pp. 18-23, 1990.

 著 者 紹 介



金 秉 燮 (準會員)

1967年 5月 5日生. 1990年 중
 앙대학교 전자공학과 공학사 학
 위취득. 1992년 2月 중앙대학교
 대학원 전자공학과 석사학위 취
 득예정. 1991년~현재 효성중공
 업 근무. 주관심분야는 Robotics,

F. A., Neural Network 등임.



沈 貴 寶 (正會員)

1956年 9月 20日生. 1984年 중
 앙대학교 전자공학과 공학사 학
 위취득. 1986년 중앙대학교 대학
 원 전자공학과 석사학위 취득.

1991년 일본 동경대학 전기·전자
 공학과 박사학위 취득. 현재 중

앙대학교 공과대학 제어계측공학과 조교수. 주관심
 분야는 Robotics, *Neural network, Fuzzy Control,
 Micro-Actuator, Sensor 등임.

◆

李 鴻 奇 (正會員) 第28卷 A編 第11號 參照
 현재 중앙대학교 제어계측공
 학과 부교수

◆

全 洪 兌 (正會員) 第26卷 第10號 參照
 현재 중앙대학교 전자공학과
 부교수