적응 학습률을 이용한 신경회로망의 학습성능개선 및 로봇 제어

Improvement of Learning Performance and Control of a Robot Manipulator Using Neural Network with Adaptive Learning Rate

이 보 희, 이 택 승, 김 진 걸
(Bo-Hee Lee, Taek-Seung Lee, Jin-Geol Kim)

Abstract : In this paper, the design and the implementation of the adaptive learning rate neural network controller for an articulate robot, which is being developed (or) has been developed in our Automatic Control Laboratory, are mainly discussed. The controller reduces software computational load via distributed processing method using multiple CPU's, and simplifies hardware structures by the time-division control with TMS320C31 DSP chip. Proposed neural network controller with adaptive learning rate structure using expert's heuristics can improve learning speed. The proposed controller verifies its superiority by comparing response characteristics of conventional controller with those of the proposed controller that are obtained from the experiments for the 5 axis vertical articulated robot. We, also, present the generalization property of proposed controller for unlearned trajectory and the change of load through experimental data.

Keywords : neural network, adaptive learning rate, TMS320C31 DSP chip, arcticulated robot

I. 서론

산업 현장 자동화의 중추적 역할을 담당하고 있는 산업용 로봇은 제어의 관점에서 볼 때 고도의 비선형적 다변수 시스템(highly nonlinear coupled multi-variable system)이며, 그 특성은 시변 비선형 2계 미분 방정식으로 표현된다. 따라서 제어 대상의 동특성을 고려하지 않는 PID등의 기구학적인 제어는 현대의 산업 현장에서 요구하는 고속, 정밀 제어와 부하 변동, 궤적 변동의 대응에는 한계가 있다.

이러한 비선형 동특성을 고려한 제어 방법으로 대표적인 토크 계산법(computed torque method)은 로봇의 동적 모델로부터 원하는 궤적에 대한 관절 토크를 연산하는 방법으로 정확한 비선형 동적 모델이 주어진 경우 훌륭한 성과를 기대할 수 있다. 그러나 Newton-Euler 알고리즘의 경우, HP-9000 마이크로컴퓨터에서 약 40ms의 시간이 걸려 실시간 구현이 어렵고, 시간 지연, 포화(Saturation), 시변 특성, 마찰 등의 정확한 비선형 동적 모델을 얻는 것이 매우 어려워 아직 실용화되지 못하고 있다[1]. 모델 기준 적응 제어(Model Reference Adaptive Control)등의 적응 제어 방법들이 이러한 문제를 해결하고자 하였으나, 미지의 파라미터 수가 증가함에 따라 복잡성이 기하급수적으로 늘어나고, 광범위한 불확실성에 대해 강인(robust)하지 못하며, 조건적으로만 안정한 문제점들이 있다.

최근 이러한 제반 문제점들을 해결하기 위해 신경회로망등 지능제어 방법을 로봇 제어에 응용하는 방법들이 연구되고 있다. 신경회로망을 이용한 로봇 제어에는 역전파(Back Propagation) 신경회로망, CMAC (Cerebellar Model Articulated Controller), Hopfield 신경회로망 등이 이용되고 있다. Seul Jung[2][3]과 Qing Li[4]는 신경회로망 제어기를 직교 좌표 공간에서 기준 궤적을 변형하는 방법으로 학습시켜 외란 제거(disturbance rejection)와 신경회로망에 강인한 특성을 부여하고자 노력하였으나 3축 수직다관절 로봇의 모의 실험에 그쳤고, Riko Šafarič[5]과 Ahmad[6]는 Lyapunov정리를 이용해 전연결, 정방행렬 가중치를 갖는 신경회로망의 안정성을 해석하였으나 전술한 특정한 구조에만 국한되었다. 또한 Mills[7]는 History Stack을 이용한 모델 기준 적응(adaptive model base) 신경회로망 제어기를 제안하였으나 이미 시간축(temporal) 학습법과 재귀(recurrent) 신경망이 개발되어 있어 큰 차이를 찾을 수 없다. 한편 Miller[8]는 공급된 토크와 출력의 관계를 이용하여, 가중치를 조정하는 CMAC 방법을 제안, 실시간 이족 보행 로봇의 제어에 적용하였다. Kar[9]은 여유 자유도를 갖는 로봇의 궤적 계획에 학습된 신경회로망을 적용하였으나 3축 평면 로봇에 대한 모의 실험에 그쳤고, Tsutumi[10]는 Hop-field 신경회로망을 이용한 8축 평면 로봇의 충돌 회피 알고리즘을 제안했다. Kawato[11]는 로봇의 역 동력학 모델을 궤환 오차 학습으로 구하는 신경회로망을 제안하였으며, 오 세영[12]은 로봇의 구조를 완전히 모른다고 가정하고, 신경회로망과 PD제어기를 병렬로 구성, 3축 로봇에 대한 실험을 하였다. 그러나 이러한 연구들은 아직까지 실험실에서 모의 실험을 하는 데 그치고 있고, 실험을 하여도 여러 가지 제약 때문에 2축, 혹은 3축에 대해서만 실험이 이루어 졌다.

본 논문에서는 신경회로망의 새로운 학습 방법을 제시하고, 본 연구실에서 개발한 5축 수직다관절 로봇에 적용, 공간상에서의 궤적 제어에 대한 실험결과와, 기존의 제어기를 사용한 경우의 결과를 비교하여 그 우수성을 확인하였다.

제안한 적응 학습률 신경회로망은 학습 성과에 따라 학습률과 가중치를 변화시켜, 보다 빠른 속도의 학습을 가능하게 하고, 극소점에서 계속되는 학습에 의한 고주파적 출력 특성을 개선하기 위해 오차의 한계치를 설정하고, 학습에 의해 한계치 이하로 오차가 줄면 학습을 중지

접수일자 : 1996. 4. 24., 수정완료 : 1996. 12. 8.
이보희 : 세명대학교 전기공학과
이택승 : LG산전 연구소
김진걸 : 인하대학교 자동화공학과
※ 본연구는 1996년도 인하대학교 연구비 지원에 의하여 1996년 연구년 기간중 수행되었습니다.

하도록 하여, 신경회로망의 학습에 적극적으로 개입하는 방법을 제안한다. 학습률을 개선할 때, 다양한 퍼지 테이블을 사용하는 퍼지 제어 방식을 사용할 수 있으나, 잦은 학습률의 변화는 오히려 학습의 에러를 증가시키는 결과를 초래하므로, 결과적으로 학습을 저하 시켜서 제작된 로봇의 시스템의 궤적 추종 특성을 저하 시킬 수 있다. 따라서 본 논문에서는 전문가의 경험 규칙을 이용하여 학습률을 변화시켜서 최소한의 학습률의 변화만을 가지는 적응학습률 구조를 만들었다. 전문가의 적응학습률 구조를 사용하는 것은 퍼지 테이블을 변환하는 과정이 별도로 필요하지 않기 때문에, 제어 방법을 간략화 할 수 있어서 실시간 제어를 할 수 있기 때문이다. 또한 제안한 제어기가 학습에 사용된 궤적뿐만 아니고 로봇의 동특성도 학습하는 일반화(generalization)특성을 가짐을 학습에 사용되지 않은 궤적과 상황에 대한 실험 결과로 제시한다. 이러한 결과는 신경회로망을 이용한 실시간 로봇의 지능 제어가 구현될 수 있음을 보여준다.

본 논문은 2장에서 역전파 신경회로망의 일반적인 구조와 학습 방법을 알아보고, 3장에서 그러한 신경회로망에 전문가의 경험지식을 적용, 학습 속도와 출력 특성을 개선하는 방법을 서술한다. 4장에서는 제어 대상인 5축 로봇과 전체 제어 구조를 살펴보고, 5장에서는 기존의 신경회로망과 제안된 구조의 제어기를 비교하고, 다양한 종류의 상황 속에서 제안된 제어기의 실험 결과를 수록하였다. 6장에서는 제안된 제어기의 성능을 평가하고 향후 연구 방향을 제시하였다.

Ⅱ. 신경회로망

1. 신경회로망의 구조

본 연구에 사용된 전연결(fully connected) 역전파 신경회로망의 일반적인 구조는 다음과 같다[13].

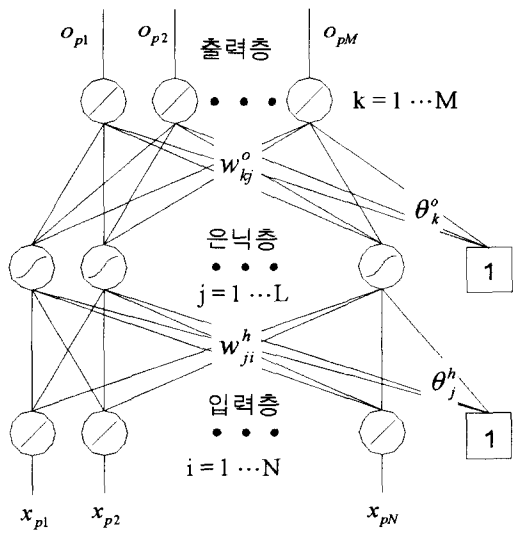

그림 1. 신경회로망의 구조.

Fig. 1. Structure of neural network.

여기에서 각 변수는 다음과 같이 정의된다.

L, N, M : 각 층의 신경 세포의 수

$\mathbf{x}_p = (x_{p1}, x_{p2}, \cdot\cdot\cdot, x_{pN})^T$: p번째 입력 벡터

w^h_{ji} : j 번째 은닉 신경 세포와 i 번째 입력 신경 세포사이의 은닉층 가중치

w^o_{kj} : k 번째 출력 신경 세포와 j 번째 은닉 신경 세포사이의 출력층 가중치

θ^h_j : 은닉층 j 번째 신경 세포의 바이어스(bias)

θ^o_k : 출력층 k 번째 신경 세포의 바이어스(bias)

o_{pk} : p번째 입력벡터에 대한 출력층 k 번째 신경 세포의 출력값

따라서 은닉층의 j 번째 신경 세포에 공급되는 실 입력 (net^h_{pj})은 다음과 같이 표현되고

$$\mathrm{net}^h_{pj} = \sum_{i=1}^{N} w^h_{ji}\, x_{pi} + \theta^h_j \tag{1}$$

은닉층의 j 번째 세포의 출력(i_{pj})과 은닉층의 출력함수 (f^h_j)는 다음과 같은 시그모이드(Sigmoid)함수로 표현할 수 있다.

$$\begin{aligned} i_{pj} &= f^h_j(\mathrm{net}^h_{pj}) \\ &= 2/(1+\exp(-\mathrm{net}^h_{pj})) - 1 \end{aligned} \tag{2}$$

은닉층과 마찬가지로 출력층의 k 번째 신경 세포에 공급되는 실 입력은 다음 식으로 표현할 수 있으며,

$$\mathrm{net}^o_{pk} = \sum_{j=1}^{L} w^o_{kj}\, i_{pj} + \theta^o_k \tag{3}$$

출력층의 k 번째 신경 세포의 출력은 선형함수의 형태로 정의한다.

$$o_{pk} = f^o_k(\mathrm{net}^o_{pk}) = \mathrm{net}^o_{pk} \tag{4}$$

여기서, 입력과 출력층의 출력함수로 선형함수를 사용하는 이유는 로봇 제어의 특성상 입력과 출력의 정확한 범위를 미리 정할 수 없고, 또한 입출력 값이 비선형 맵핑되면 오차의 정확한 반영과 제어기의 출력이 로봇에 정확히 공급될 수 없기 때문이다.

바이어스는 출력이 항상 1인 신경세포로 바이어스의 가중치는 일반 가중치와 동일한 방법으로 학습된다. θ^h_j 과 θ^o_k의 바이어스항의 추가는 출력함수(선형 함수와 시그모이드 함수)의 특성상 입력이 영이면 출력 또한 영이외의 값은 출력될 수 없다. 그러나 로봇의 제어에는 중력항등 비선형 항이 존재하므로, 입력이 영이라도 출력은 영이 아니어야 하는 상황이 존재한다. 따라서 이러한 바이어스항의 첨가로 바이어스된 형태의 입력과 출력의 비선형 맵핑의 학습이 가능하게 되는 것이다.

2. 일반화된 델타 규칙(Generalized delta rule)

여기서는 전술한 일반적인 신경회로망이 어떻게 학습되어 지는 가를 살펴 본다.

먼저 학습을 위한 목표 오차 함수를 다음과 같이 정의한다.

$$E_p = \frac{1}{2}\sum_{k=1}^{M}(y_{pk} - o_{pk})^2 \tag{5}$$

여기서, y_{pk}는 원하는 p번째 입력벡터에 대한 k번째 출력층 신경세포의 출력이고, o_{pk}는 실제의 p번째 입력벡터에 대한 k번째 출력층 신경세포의 출력이며, M 은 출력층의 신경세포의 수이다. 상수 1/2은 수학적으로 다루기 편리하도록 첨가한 것이다.

이러한 목표 오차 함수를 최소화하기 위해 최급강하법

(steepest descent method)을 사용한다. 최급강하법이란 최소화하기를 원하는 함수의 구배(gradient)를 구하고 구배의 반대 방향으로 변수를 이동시키는 것이다. 즉, 목표함수를 출력층의 가중치로 연쇄 규칙(chain rule)을 적용, 편미분하며 (6)과 같다

$$\frac{\partial E_p}{\partial w_{kj}^o} = -(y_{pk} - o_{pk}) \frac{\partial f_k^o(\mathrm{net}_{pk}^o)}{\partial(\mathrm{net}_{pk}^o)} \frac{\partial(\mathrm{net}_{pk}^o)}{\partial w_{kj}^o} \tag{6}$$

이때의 출력층의 가중치 조정식을 표현하면 다음과 같다[13].

$$w_{kj}^o(t+1) = w_{kj}^o(t) + \eta\, \delta_{pk}^o\, i_{pj} \tag{7}$$

여기서, η 는 학습률을 결정하는 상수로 일반적으로 0.1~0.9사이의 값을 사용하며 ,출력층의 델타항은 (8)과 같다.

$$\begin{aligned} \delta_{pk}^o &= (y_{pk} - o_{pk}) f_k^{o\prime}(\mathrm{net}_{pk}^o) \\ &= (y_{pk} - o_{pk}) \end{aligned} \tag{8}$$

비슷한 방법으로 은닉층의 가중치 및 은닉층의 델타항을 구하면 (9)와 (10)과 같다[13].

$$w_{ji}^h(t+1) = w_{ji}^h(t) + \eta\, \delta_{pj}^h\, x_i \tag{9}$$

$$\delta_{pj}^h = (1 - o_{pk}^2) \sum_k \delta_{pk}^o\, w_{kj}^o \tag{10}$$

이 과정을 보다 개념적으로 설명하면 출력층의 오차는 각 가중치 만큼씩 은닉층 신경세포의 출력에 원인이 있으므로 출력층의 오차를 가중치 만큼씩 은닉층에 나누어 주고 이것을 은닉층의 가중치로 편미분, 구배를 구해 방향을 결정하고 이것을 은닉층의 델타항이라 한다. 전달받은 델타항을 가지고 출력층의 가중치 조정과 같은 방법으로 은닉층의 가중치를 조정하는 것이다[14].

Ⅲ. 신경회로망의 개선

1. 지역극소(Local minimum)의 탈출

이러한 일반적인 신경회로망의 학습방법은 지역 극소가 있는 경우 문제가 발생할 수 있다. 즉 지역 극소에 빠지면 전역 극소를 찾아 탈출하는 것이 불가능하다. 이러한 문제를 해결하기 위해 관성항(Momentum)을 첨가한다. 관성항이란 (7)과 (9)에 과거의 가중치변화량을 일정비율 더하여 줌으로서 마치 관성이 존재하는 것처럼 동작하게 하는 것이다. 이를 정리하면 (11)과 (12)와 같다.

$$\begin{aligned} w_{kj}^o(t+1) = w_{kj}^o(t) &+ \eta\, \delta_{pk}^o\, i_{pj} \\ &+ \underline{\alpha\, \Delta w_{kj}^o(t-1)} \end{aligned} \tag{11}$$

$$\begin{aligned} w_{ji}^h(t+1) = w_{ji}^h(t) &+ \eta\, \delta_{pj}^h\, x_i \\ &+ \underline{\alpha\, \Delta w_{ji}^h(t-1)} \end{aligned} \tag{12}$$

여기서, α는 관성항 계수로 일반적으로 0.5~0.95사이의 값을 사용하며, $\Delta w_{ji}^h(t-1)$와 $\Delta w_{kj}^o(t-1)$는 각각 관성항을 제외한 과거의 가중치 변화량이다[13].

그러나 관성항이 모든 지역 극소를 탈출할 수 있도록 해 주는 것은 아니어서 α값이 작을 경우 결국 지역극소에서 벗어나지 못하는 경우도 있다. 따라서 이러한 α값의 선정은 시행착오를 거쳐 각각의 경우에 맞는 적당한 값을 찾아야 한다. 그러나 이것 또한 신경회로망이 전역극소를 찾을 가능성을 높인 것이지 모든 지역극소를 탈출하여 항상 전역극소에 도달하는 것은 아니다. 이는 뒤에 제안할 전문가의 경험적 지식을 도입함으로서 보다 전역극소를 찾을 가능성을 보다 높일 수 있다.

2. 전문가의 경험(heuristics) 도입

신경회로망의 학습 과정에 직접 개입하여 학습률과 관성항을 상황에 맞게 조정할 수 있다면 보다 나은 결과를 얻을 수 있다. 적응 학습률(adaptive learning rate)이란 이런 근거를 바탕으로 고안되었다. 다음은 본 실험에 적용한 규칙들이다.

규칙 1 : IF 새로운 오차 > 과거의 오차 * 1.04
THEN 새로운 학습률 = 과거의 학습률 * 0.7,
새로운 관성항 = 과거의 관성항 * 0.7

여기서, 새로운 오차 및 과거의 오차는 샘플링 구 간에서의 오차, 즉 e(k), e(k-1)을 의미한다. 규칙 1은 학습 과정 중 학습의 방향이 잘못되어 오차가 커져 가는 경우에 방향의 전환을 빠르게 하여 준다. 4 %의 여유를 둔 것은 관성항의 영향으로 지역극소를 벗어나는 것을 이 규칙이 방해하지 않도록 한 것으로 30%의 학습률과 관성항의 감소와 같이 실험적 경험을 바탕으로 설정되었다.

규칙 2 : IF 새로운 오차 < 과거의 오차
THEN 새로운 학습률 = 과거의 학습률 * 1.05,
새로운 관성항 = 과거의 관성항 * 1.05

규칙 2는 학습의 방향이 잡히면 학습의 속도를 증가시켜 보다 빨리 극소에 도달할 수 있도록 한 것으로 역시 5 %의 증가율은 실험적 경험에 의한 것이다. 이것은 또한 관성항을 키워 앞에서 보았던 지역극소의 탈출 가능성을 높여준다.

규칙 3 : IF 새로운 오차 < 허용 오차(E_p = 1.5)
THEN 새로운 학습률 = 0,
새로운 관성항 = 0

관성항을 가지고 있는 신경회로망은 극소에서 관성항의 영향으로 진동하는 경향이 있다. 또한 실제 로봇에 적용한 결과, 기어의 백래쉬(back lash)는 이러한 진동을 더욱 증폭시킬 뿐만 아니라, 이러한 고주파적 출력특성은 시스템의 안정성에 결정적인 영향을 줄 수 있는 전자적 잡음의 발생 가능성을 높인다. 규칙 3은 이러한 신경회로망의 극소점에서의 학습에 의한 고주파적 출력 특성을 개선하기 위해 첨가 된 것으로 오차가 미리 정의한 허용한도 이하로 내려가면 학습을 중단시켜 로봇의 제어에서는 치명적일 수 있는 진동을 미연에 방지하는 기능을 한다.

그러나 규칙 1의 4%의 여유나, 규칙 3의 허용 오차 1.5의 값은 적용할 시스템에 따라 변동의 가능성이 있고, 또한 이러한 규칙들로 개선된 신경회로망도 모든 지역극소를 탈출 할 수 있는 것은 아니며, 전역극소가 가중치의 변화 범위, 즉 CPU안에서 표현 가능한 수형의 범위 안에 없는 경우나, 전역극소쪽의 기울기(ascent)가 매우 커서 관성항만으로는 넘을 수 없는 경우, 또한 허용오차 한도를 만족시킬 수 있는 전역 극소가 없는 경우, 반대로 허용 오차 한도를 만족시키는 경우가 여러 개 있는 경우 등 문제가 일어날 수 있는 경우가 여러 가지가 있고, 또 이를 해결하기 위한 시뮬레이티드 어닐링(simulated annealing), 볼츠만 머신(boltzmann machine), 확률적 구배 탐색(stochastic gradient search), 유전 알고리즘(genetic algorithms)의 도입 등과 같은 신경회로망의 새로운 구조와 학습방법이 개발되어 있다[15]. 하지만 이

러한 방법들은 아직까지는 실시간 제어 환경에서 학습과 제어를 동시에 하는 구조로 구현될 수 없고, 주 제어기의 설계상 32Kbyte의 메모리안에 구현하기에는 무리가 따른다.

Ⅳ. 제어 대상과 제어 구조

1. 제어 대상

실험에 사용한 제어 대상은 본 연구실에서 개발한 LPR 593 5축 수직 다관절 로봇이다(그림 2). 본 로봇은 5개의 LG 산전 65V 100W DC 서보 모터로 구동된다. 각각의 서보 모터는 20 Khz 펄스폭 변조 방식으로 구동되며, 1000 Pulse/Rev.의 분해능을 가진 증분형 엔코더를 위치 센서로 사용하고 있다. 또한 각 축의 서보 모터는 허리축(waist)부터 차례로 1/70, 1/264, 1/180, 1/60, 1/60의 감속기를 통해 링크와 연결되어 있다. 감속기는 유성치차열을 사용한 방식으로 고토크 전달 및 소형 경량화하였다. 각 관절부의 구동에 의한 영향을 최소화하기 위하여 기저부에 전체 자중의 60%가 집중되도록 설계하였다. 또한 제어의 관점에서 보면 코리올리힘에 의한 토크항의 상쇄와 토크 계산식의 비선형항을 제거할 수 있는 4절 링크 형상[16]을 택하여 설계하였다.

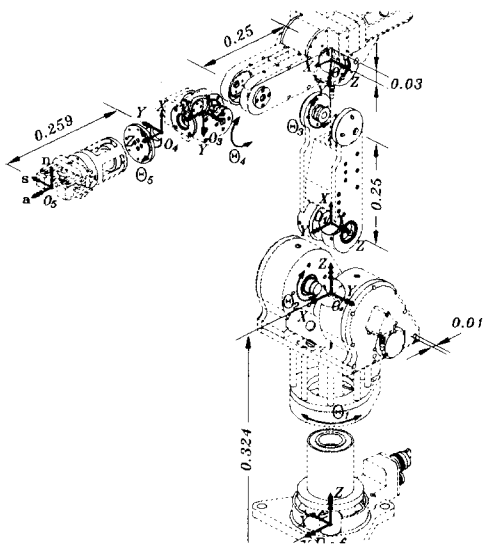

그림 2. LPR 593 수직다관절로보트(단위 : m).
Fig. 2. LPR 593 articulated robot.

각 관절의 축은 가운데를 비우고 테이퍼(taper) 형상을 갖도록 하여 미소 편심을 제거하고 배선 시 편리하게 했다. 손목 관절은 토크 이득 및 작동 범위의 확장을 고려하여 차동치차열을 이용하여 구동 되도록 설계하였으며, 구동원은 어깨 및 팔꿈치 관절 및 부하를 줄이기 위해 팔꿈치 관절 상에 위치하여 타이밍 벨트를 이용하여 원격 구동할 수 있게 했다. 또한 어깨 구동부와 팔꿈치 구동부에는 전자 제동 장치를 장착하여 정전시와 비상시에 대한 대비를 하였다. 허리 구동부는 도립형상으로도 작업이 가능하도록 테이퍼 로울러 베어링을 이용하여 축방향 하중에 견딜 수 있도록 하였으며, 팔꿈치 관절에 오프셋을 주어 작업 반경이 증대 되도록 하였다. 로보트의 전체 사양은 표 1과 같다.

2. 제어 구조

본 연구에 사용된 전체 시스템의 구성은 그림 3과 같이 온라인 시뮬레이터(On-line Simulator), 주 제어기, 서보 모터 구동기, 엔코더 신호 해독기, 그리고 LPR 593 로봇의 다섯 부분으로 크게 나눌 수 있다. 제안된 제어기는 기본적으로는 PD제어기를 사용하였고 신경회로망을 병렬로 사용하여 보완한 구조를 사용하였다. 따라서 신경회로망이 초기 학습 방향을 잡아주며, 신경회로망이 충분히 학습된 후에는 거의 영향을 미치지 않고 PD에 의한 영향을 받을 수 있게 하였다.

표 1. 로봇 사양.
Table 1. Robot specification.

형 상 Construction	수직다관절(밀폐형) Vertical articulated arm (Closed structure)		반복정밀도 Repeatability	± 0.1 mm	
자 유 도 Degree of Freedom	5축과 공구부 5axes plus end effector		구 동 원 Actuators	직류서보모터 DC servo motor	
도달거리 Reach (Wrist Centre)	508.5mm		중 량 Weight	66.9kg	
관절작동범위 Joint Operation Range	Base	356° (±178°)	최소증분량 Incremental Movement	Base	17.45×10e-6rad
	Shoulder	180° (+90°)		Shoulder	5.83×10e-6rad
	Elbow	140° (+70°)		Elbow	9.52×10e-6rad
	Pitch	200° (+100°)		Pitch	39.27×10e-6rad
	Roll	356° (±178°)		Roll	39.27×10e-6rad
정격최대속도 Maximum Speed	Base	200 deg/sec	가반하중 Pay Load	유효 Effective	3.0kg (6.6lbs)
	Shoulder	66.79 deg/sec			
	Elbow	109.09 deg/sec			
	Pitch	450 deg/sec		최대 Maximum	3.5kg (7.7lbs)
	Roll	450 deg/sec			

2.1 온라인 시뮬레이터

온라인 시뮬레이터는 80486DX2-66 IBM PC 호환기종에 RS-422 통신보드를 장착하고 있으며, 자체 제작한 로봇 언어를 이용한 궤적 계획, 정기구학 및 역기구학 계산을 수행한다. 또한 계산된 관절각을 주 제어기로 송신하는 기능 및 주 제어기에서 실제로 계측된 관절의 위치를 수신하여 파일로 저장하고 모니터 상에 실제의 로봇 자세를 표시하는 기능을 한다. 궤적 계획은 5차 다항식을 이용하여 궤적을 생성을 한 후, 역기구학을 이용하여 관절의 움직임을 계산하고, 주 제어기와 연결하지 않은 채, 모니터 상에서 실행시켜 실제 온라인 동작 시 발생할 수 있는 에러를 줄일 수 있게 구성하였다. 매 20ms마다 연산된 각 관절의 위치 명령 값이 57600 bps의 RS-422 방식으로 주제어기에 송신되고, 주 제어기로부터 현재의 관절각을 받아 정기구학을 해석하여 로봇의 실제 운동 상황을 모니터 상에 실시간으로 표시한다. 또한 각 관절의 실측치를 실시간으로 모니터에 표시함으로써 실제 관절의 움직임을 알기 쉽게 하였고, 파일로 저장하여 관절의 움직임을 분석할 수 있게 하였다.

2.2 주 제어기

주 제어기는 서보 제어기의 TMS320C31 DSP CPU와, 입/출력 제어기와 RS-422 통신 제어기의 역할을 수행하는 V25 CPU, 두 CPU사이의 통신을 담당하는 양방향(dual port) 메모리 등으로 구성되며, 온라인 시뮬레이터에서 계획된 관절각을 받아 각 축의 서보 모터 구동 및 제어를 수행한다. 또한 엔코더를 이용하여 관절각을 계측하여 온라인 시뮬레이터로 송신하는 기능을 가지고 있다. 서보 제어기는 각 관절을 동시(향후 확장을 위하여 8개의 서보모터를 제어 할 수 있도록 설계되었음.)에 일괄 제어하여야 하는데 기존의 범용 마이크로 프로세서로는 실시

간 구현이 불가능하므로 본 제어기에서는 TI사의 부동소수점 DSP(Floating Point Digital Signal Processor)인 TMS320C31-33Mhz을 사용하여 시분할 방식에 의해 각 관절을 제어함으로써 기존의 각 관절마다 한 개씩의 마이크로 프로세서가 담당하는 복잡한 방식의 하드웨어 및 소프트웨어 구조를 대폭 간소화하였다. TMS320C31은 외부에 32Kbyte, 15ns의 SRAM을 주 메모리로 사용하여 다양한 제어 알고리즘을 사용할 수 있게 하였으며 전체 주 제어기의 동작은 그림 4와 같다

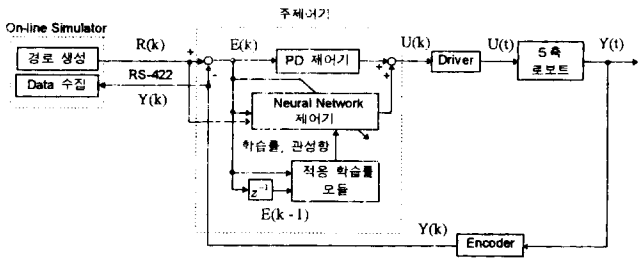

그림 3. 전체 시스템의 구조.

Fig. 3. Total system structure.

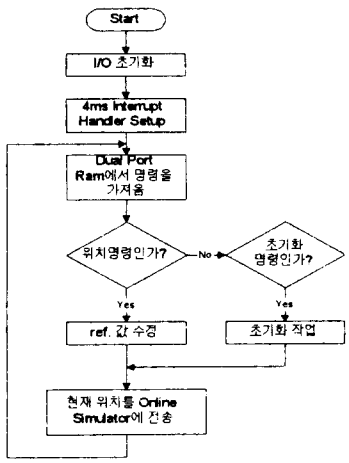

그림 4. 주제어기의 흐름도.

Fig. 4. Flow chart of main controller.

2.3 신경회로망 제어기

본 논문에서 제안한 알고리즘은 어셈블러를 삽입한 (in-line assembler) C 언어를 이용해 구현하였다. 주 제어기는 DSP CPU안에 PD제어기와 신경회로망의 병렬 구조로 되어 있으며 신경회로망의 입력층은 각 축의 기준치, 오차, 오차의 미분항을 입력받는 15개의 신경세포로 구성되어 있다. 은닉층에는 7개의 신경세포가 있으며 시그모이드 함수가 출력 함수로 사용된다. 마지막으로 출력층에는 5개의 신경세포가 로봇의 각 축마다 하나씩 배당되어 있다. 바이어스 항을 포함하여 총 152개의 가중치가 매 4ms의 샘플링 시간마다 입력에 상응하는 출력을 내보내고, 오차에 의해 학습된다. 학습률의 초기치는 0.3이고, 관성항의 초기치는 0.9이다.

학습에 사용되어지는 오차항은 로봇을 원하는 위치로 이동시키기 위한 정확한 신경회로망의 출력(y_{pk})값을 알 수가 없으므로 각 축의 오차를 해당 출력 신경세포의 오차로 사용한다. 이것은 로봇의 동역학적 특성뿐만 아니라 주변 제어회로의 특성까지 모두를 학습할 수 있게 한다[12].

양방향 메모리로부터 궤적 계획된 관절각을 가져오고, 신경회로망의 출력을 구하고, 가중치를 조정하며, 보조 제어기인 PD 제어기의 출력을 구하고, 엔코더 신호를 해독하고, 제어신호를 내보내는 모든 과정은 DSP CPU에서 약 3ms의 시간이 소요된다.

따라서 샘플링 주기는 4ms로 결정하였으며 전체 제어알고리즘은 그림5와 같다. 4ms는 서보 모터와 로봇의 대역폭(band width)을 충분히 벗어난 값으로 제어시스템은 아날로그 시스템으로 해석될 수 있다.

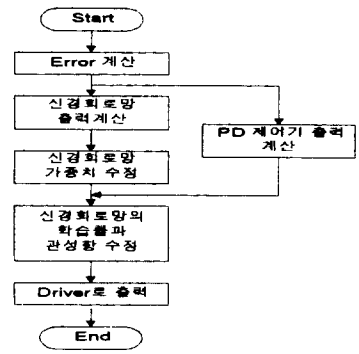

그림 5. 신경회로망 제어기의 흐름도.

Fig. 5. Flow chart of NN controller.

V. 실험 결과

1. 기존의 제어기와 비교

1.1 PD 제어기

그림 6와 그림 7은 신경회로망 없이 PD제어기만으로 직경 10cm의 공간상의 원형 궤적에 대하여 제어를 수행한 것이다.

평면도와 정면도를 로봇의 기준 좌표계를 기준으로 표현하였다. 전술하였듯이 로봇이 수행한 결과를 엔코더를 통해 온라인 시뮬레이터에서 파일로 저장하고 역기구학

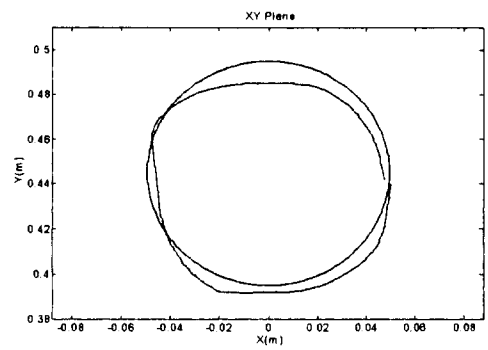

그림 6. PD 제어시의 평면도.

Fig. 6. Plane figure with PD control.

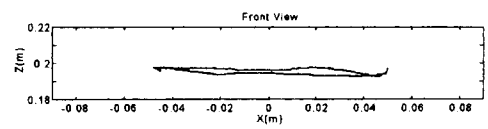

그림 7. PD 제어시의 정면도.

Fig. 7. Front figure with PD control.

을 통해 공간상의 직교좌표계로 변환한 것이다. 최대 속도는 0.1 $m/\sec$ 이며 최대 가속도는 0.05 $m/\sec^2$ 이고 총 수행 시간은 7.2 초가 걸렸다. PD제어기의 이득은 계단입력에 대해 0.4초 정도의 정착시간과 오버슈트가 없는 오버댐핑의 특성을 갖도록 조정한 후 본 실험 궤적에 적용

한 결과이다.

1.2 기존의 신경회로망 제어기

그림 8과 그림 9는 기존의 고정 학습률 신경회로망과 PD제어기로 앞에서와 마찬가지로 직경 10cm의 공간상의 원형 궤적에 대하여 제어를 수행한 것이다.

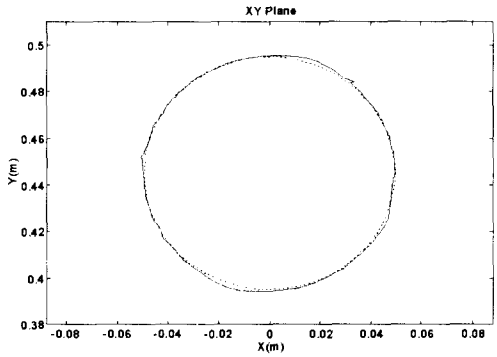

그림 8. 고정 학습률 신경회로망 제어시의 평면도.

Fig. 8. Plane figure with Fixed learning rate NN control.

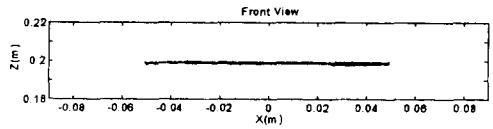

그림 9. 고정 학습률 신경회로망 제어시의 정면도.

Fig. 9. Front figure with fixed learning rate NN control.

고정 학습률은 뒤에 나올 적응학습률 신경회로망의 초기치와 같은 0.3이며, 관성항 역시 적응학습률 신경회로망의 초기치와 같은 0.9로 하여 온라인으로 학습하며 첫 번째 시도한 경우이다. 이하 조건은 앞에서의 경우와 같다. 앞의 PD제어기만으로 제어를 한 경우에 비해 학습에 의한 보상의 효과를 확실히 보여주고 있다. 정면도에서 2mm정도의 Z축 방향의 오차가 보이고, 평면도에서 원의 좌측 하단과 우측상단에서 2mm정도의 직교 좌표계 오차가 발생하고 있다. 이러한 결과는 학습을 계속함으로서, 즉 같은 궤적에 대해 반복적으로 제어를 수행함으로서 좀 더 나은 결과를 얻을 수 있다.

1.3 기존 신경회로망 제어기의 고주파 출력 특성

그림 10은 기존의 고정 학습률 신경회로망과 PD제어기로 앞에서와 마찬가지로 직경 10cm의 공간상의 원형 궤적에 대하여 다섯 번의 제어를 반복 시도한 후의 결과이다. 반복적으로 제어를 수행함으로서 좀 더 나은 결과를 얻고자 하였으나 가중치의 초기치가 잘못되어 적절한 학습이 이루어지지 못하고 기어의 백래쉬와 극소 근처에서 반복되는 학습으로 인한 고주파적 출력특성이 궤적 전반에 걸쳐 보인다.

이하 조건은 앞에서의 경우와 같다. 이러한 고주파적인 출력 특성은 제어의 관점에서 볼 때 제어 에너지의 과도한 출력으로 비효율적일 뿐만 아니라, 동역학적으로는 진동으로 인한 로봇의 기계적 충격과 소음 등이 발생하고, 또한 전자적인 잡음의 발생 가능성을 높인다. 이러한 전자적 잡음의 발생은 엔코더 회로부나 제어용 CPU 회로에 침투할 경우 시스템 전체의 안정성은 물론 사고의 원인이 될 수 있다. 따라서 반드시 제거되어야 할 사항이다.

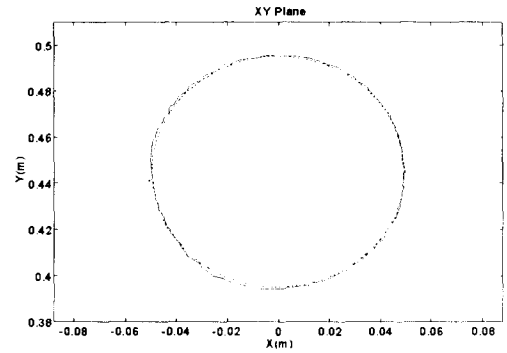

그림 10. 고정 학습률 신경회로망 제어시의 평면도 (고주파 출력 특성).

Fig. 10. Plane figure with fixed learning rate NN control(high frequency characteristic).

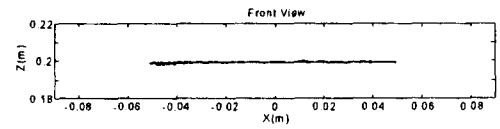

그림 11. 고정 학습률 신경회로망 제어시의 정면도 (고주파 출력 특성).

Fig. 11. Front figure with fixed learning rate NN control(high frequency chacteristic).

1.4 제안한 적응 학습률 신경회로망 제어기

그림 12는 제안한 적응 학습률 신경회로망 제어기를 이용하여 앞에서와 같은 직경 10cm의 공간상의 원형 궤적에 대하여 첫 번째 제어를 수행한 것이다.

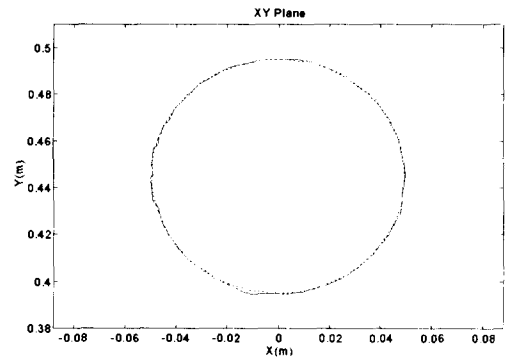

그림 12. 적응 학습률 신경회로망 제어시의 평면도.

Fig. 12. Plane figure with adaptive learning rate NN control .

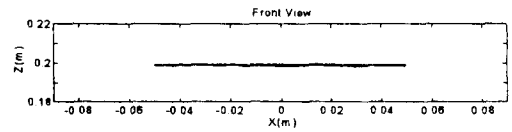

그림 13. 적응 학습률 신경회로망 제어시의 정면도.

Fig. 13. Front figure with adaptive learning late NN control.

학습률의 초기값은 0.3이고 관성항의 초기값은 0.9로, 앞의 실험에서 사용한 고정학습률 신경회로망의 것과 동일한 값을 초기값으로 설정하였다. 이하 조건은 앞에서의 경우와 같다. 기존의 고정학습률 신경회로망을 이용한 경

우보다 빠른 속도로 학습이 진행되어 첫 번째 시도에서 학습이 거의 마무리됨을 알 수 있다. 좌측 하단에서 미소한 변화를 갖는 Elbow의 움직임을 미처 따라가지 못하는 모습이 보이나 전체적으로는 적절한 제어가 수행되고 있다.

5.1. 제어기의 오차 비교

그림 14에서 그림 18까지는 PD제어기, 기존의 신경회로망 제어기(첫 번째 시도), 기존의 신경회로망 제어기(다

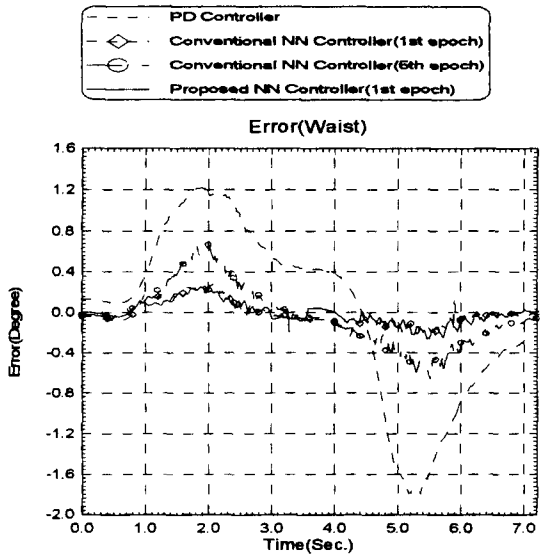

그림 14. Waist축의 오차.

Fig. 14. Waist error.

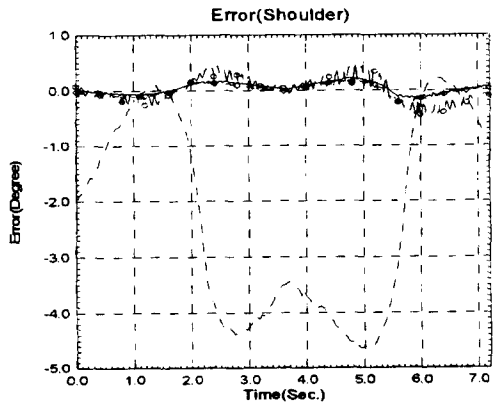

그림 15. Shoulder축의 오차.

Fig. 15. Shoulder error.

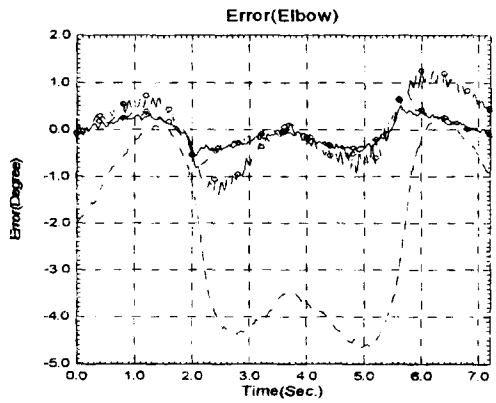

그림 16. Elbow축의 오차.

Fig. 16. Elbow error.

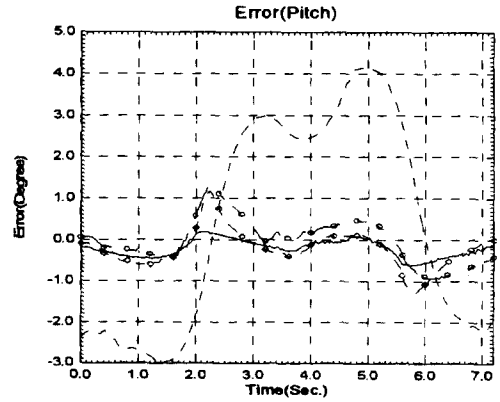

그림 17. Pitch축의 오차.

Fig. 17. Pitch error.

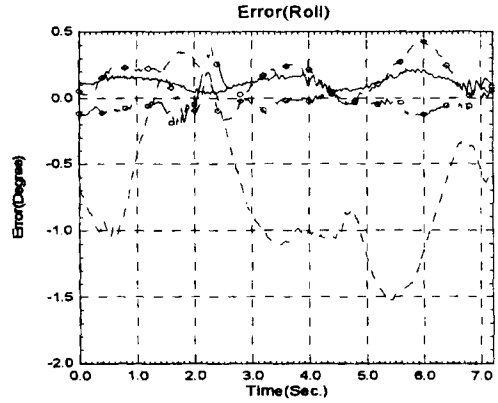

그림 18. Roll축의 오차.

Fig. 18. Roll error.

섯 번째 시도), 제안한 신경회로망 제어기의 공간상의 원형 궤적에 대한 오차를 분석한 것이다.

X축은 시간 축으로 초 단위로, Y축은 오차를 도(degree)로 표현한 것이다. Y축의 스케일은 각 축마다 오차의 양에 따라 다르게 잡았다. 각 축의 최대 오차는 각 축의 회전 방향이 바뀌거나 정지하였다 이동을 시작하는 부분에서 발생하였다. 기존의 신경회로망 제어기로 다섯 번의 반복 제어를 한 경우 Shoulder축과 Elbow축에서 진동이 발생하고 있음을 확연히 알 수 있다. 실험적으로는 이러한 진동이 매번 발생하는 것은 아니며 가중치의 초기치에 영향을 받는다. 가중치를 새로 초기화 할 때 10%정도의 발생확률을 가지고 있었다. 물론 신경세포의 수를 조정하여 발생확률을 줄이는 것도 가능하나 그렇게 하려면 적용 대상이 바뀔 때마다 매번 제어기의 구조를 시행착오를 거쳐 다시 설계해야하는 문제가 발생한다. 따라서 보다 적극적으로 대처하고자 본 논문에서 제안한 적응학습률 구조를 도입하였다.

제안한 구조의 제어기에서는 Shoulder와 Elbow에서 보이던 진동이 확실히 억제되고 있음을 알 수 있다. 다른 축에 비해서 상대적으로 기존의 신경회로망제어기가 오차를 많이 줄이지 못했던 Pitch축과 Roll축은 제안한 제어기의 도입으로 많은 양의 오차 감소가 보이고 있다. 이상의 결과를 표를 이용하여 정리하면 표 2와 같다.

제안한 제어기에서 Elbow, Pitch, Roll축의 오차가 기존의 신경회로망 제어기보다 상대적으로 큰 비율로 감소하였으며, 고주파에 의한 영향이 없으며, 진동으로 인한 오

차의 증가가 없음을 알 수 있다.

표 2. 각 제어기의 오차분석.

Table 2. Error analysis of controllers.

제어기 / 축	PD 제어기	기존 신경회로망 (첫 번째 시도)		기존 신경회로망 (다섯 번째 시도)		제안한 신경회로망 (첫 번째 시도)	
	최대오차	최대오차	PD 대비 감소율	최대오차	PD 대비 감소율	최대오차	PD 대비 감소율
Waist	-1.800°	0.293°	83.7%	0.666°	63.0%	0.263°	85.4%
Shoulder	1.476°	-0.232°	84.3%	-0.452°	69.4%	0.230°	84.4%
Elbow	-4.646°	0.824°	82.3%	-1.393°	70.0%	-0.603°	87.0%
Pitch	4.146°	-1.283°	69.1%	1.370°	67.0%	-0.626°	85.0%
Roll	-1.531°	0.442°	71.1%	0.391°	74.5%	0.218°	85.8%

5.2 제안한 적응 학습률 신경회로망의 일반화 특성

다음은 제안한 구조의 제어기의 궤적, 속도, 부하의 변동에 대한 일반화 특성을 살펴보기 위하여 몇 가지 변화를 주어 보았다.

5.2.1 작업 속도의 변화

앞에서와 마찬가지로 직경 10cm의 공간상의 원형 궤적에 대하여 최대 속도는 0.1 $m/\sec$에서 1.0 $m/\sec$로 증가시켰고, 최대 가속도는 0.05 $m/\sec^2$에서 0.2 $m/\sec^2$로 증가 시켰다. 총 수행 시간은 3.6초가 걸렸다.

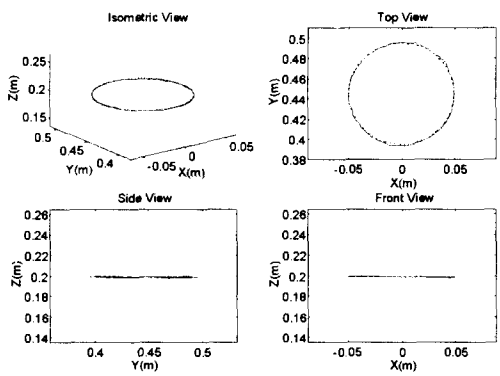

그림 19. 이동 속도 증가시의 궤적.

Fig. 19. Trajectory under the increased speed.

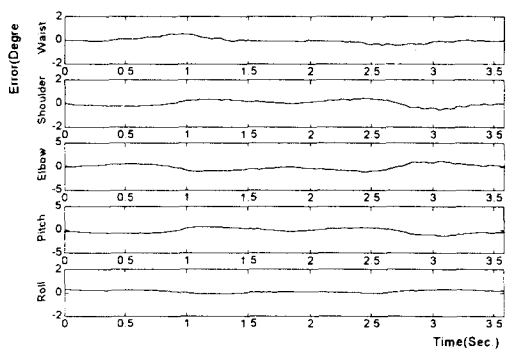

그림 20. 이동 속도 증가시 각 축의 오차.

Fig. 20. Joint errors under the increased speed.

역시 학습률 초기값은 0.3이며, 관성항 초기값은 0.9이며 앞에서 실험하여 얻은 학습된 가중치를 이용, 실험한 것으로 오차가 상대적으로 커지기는 하였으나 속도가 증가되어도 충분히 제어할 수 있음을 보여준다. 각 축의 최대 오차는 Waist는 0.543°, Shoulder는 -0.525°, Elbow는 1.281°, Pitch는 -1.267°, Roll은 0.361°이다.

5.2.2 사각형 궤적에 적용

앞에서 학습된 신경회로망이 궤적이 바뀌면 어떻게 대응하는 지를 보기 위해 이번에는 가로 세로 10cm의 공간상의 정사각형 궤적에 대하여 실험하였다. 최대 속도는 0.1 $m/\sec$, 최대 가속도는 0.05 $m/\sec^2$이다. 총 수행 시간은 16.46초가 걸렸다. 역시 학습률 초기값은 0.3이며, 관성항 초기값은 0.9의 앞에서 실험하여 얻은 학습된 가중치를 이용, 실험한 것으로 이것으로 신경회로망이 궤적에 대한 학습이 충실히 되어 있음을 알 수 있다.

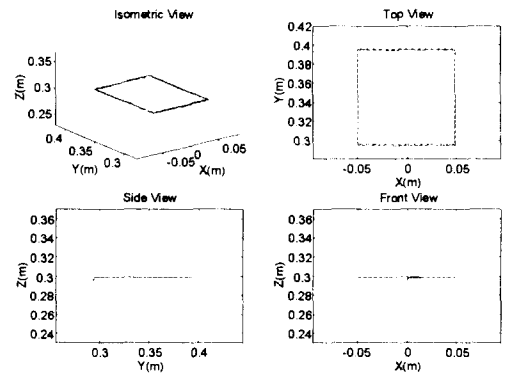

그림 21. 사각형 궤적.

Fig. 21. Rectangular trajectory.

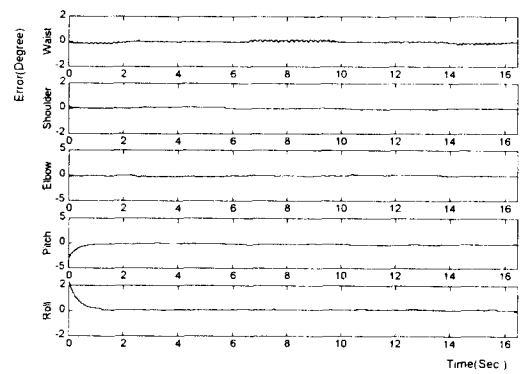

그림 22. 사각형 궤적에서의 각 축의 오차.

Fig. 22. error of rectangular trajectory.

사각형 궤적에서 각 축의 오차를 그린 것으로 Pitch와 Roll의 시작부분에서 오차가 크게 발생한 것은 궤적 계획상의 문제로 최대 토크로도 따라 갈 수 없는 시작점까지의 빠른 이동 때문이었다.

5.2.3 높이의 변화가 있는 공간상의 임의의 궤적

공간상의 임의의 궤적에 대한 실험으로 로봇의 한계 내에서 최대 속도로 위의 궤적을 수행하도록 한 것으로 Z축 방향의 높이 변화와, 직선이 아닌 3차원 공간상의 곡선으로 점들 사이의 궤적을 주었으며, 총 수행 시간은 3.5초이다. 역시 제안한 구조의 신경회로망이 주어진 작업의 종류가 크게 변하여도 충실히 제어를 수행함을 알 수 있다. 가속도가 급격히 변하는 모서리 부분에서 오차가 발생함을 알 수 있으나 전반적으로 만족할 만한 결과를 보이고 있다.

그림 24는 각 축의 오차를 그린 것으로 가속도가 급변하는 부분에서 오차가 크나 이동 속도를 감안 할 때 큰 무리 없이 궤적에 대한 제어를 수행함을 알 수 있다.

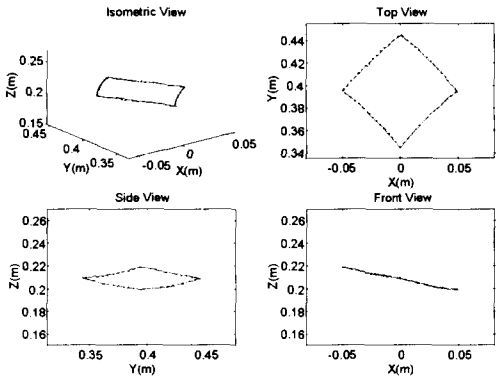

그림 23. 공간상의 임의의 궤적.

Fig. 23. Trajectory in 3D space.

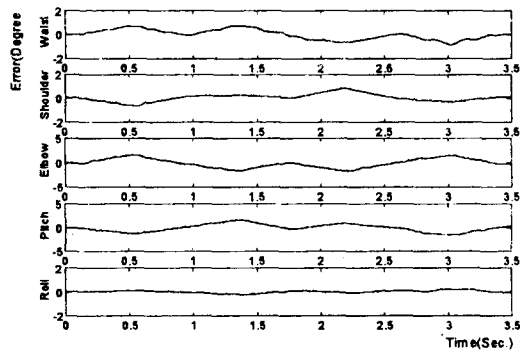

그림 24. 공간상의 임의의 궤적에서의 각 축의 오차.

Fig. 24. Joint trajectory errors in 3D space.

5.2.4 부하의 변동

앞에서와 같이 직경 10cm의 공간상의 원형 궤적에 대하여 최대 속도는 0.1 $m/\sec$, 최대 가속도는 0.05 $m/\sec^2$로 하고, 로봇의 그립퍼에 2.5Kg의 부하를 걸어 제어를 수행하였다. 총 수행 시간은 7.2초가 걸렸다. 실험에 사용한 로봇의 최대 가반하중은 3.5Kg, 유효 가반하중은 3Kg으로 설계되었다.

역시 앞에서 실험하여 얻은 학습된 가중치를 이용, 실험한 것으로 부하가 없을 때와 비교하여 오차의 변화는 매우 작은 양으로 가반하중내의 부하변동에 충분히 대응할 수 있음을 보여준다.

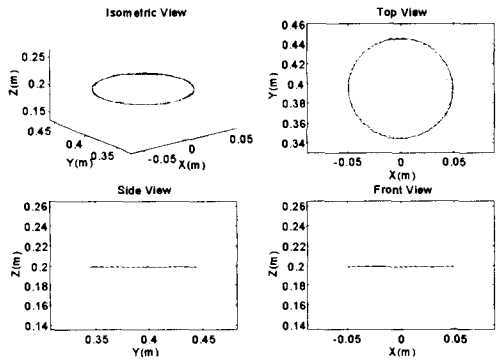

그림 25. 부하 변동시의 궤적.

Fig. 25. Trajectory under load variation.

역시 앞에서 실험하여 얻은 학습된 가중치를 이용, 실험한 것으로 부하가 없을 때와 비교하여 오차의 변화는 매우 작은 양으로 가반하중내의 부하변동에 충분히 대응할 수 있음을 보여준다.

각 축의 최대 오차를 부하가 없을 때와 비교해 보면 Waist는 0.263°에서 0.6140°로, Shoulder는 0.230°에서 0.205°로, Elbow는 -0.603°에서 -0.663°로, Pitch는 -0.626°에서 0.835°로, Roll은 0.218°에서 -0.247°로 2.5Kg의 부하가 있어도 오차의 큰 증가 없이 제어를 수행하고 있다.

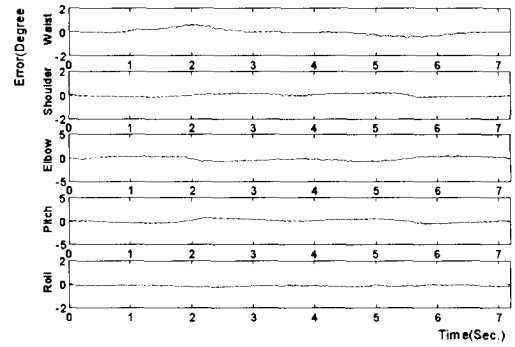

그림 26. 부하 변동시 각 축의 오차.

Fig. 26. Joint errors under load variation.

VI. 결론

본 논문에서는 기존의 PD제어와는 달리 다양한 궤적의 변화와 부하의 변동에 대응할 수 있는 적응 학습률을 이용한 신경회로망을 제안하였다. 따라서 제어대상의 구조적인 지식 없이도 최소화해야 할 오차항의 정의만으로 제어가 가능했고, 제어대상과 제어시스템에 대한 모델링도 필요없었다. 또한 기존의 신경회로망보다도 학습속도와 제어 성능 면에서 개선이 있었음을 알 수 있다. 제안한 신경회로망은 학습하지 않은 궤적이나 속도의 변동, 부하의 변동 등에도 훌륭히 제어를 수행하며, 기어의 백래쉬와 극소 근처에서 반복되는 학습으로 인한 고주파적 출력 특성을 극복할수 있음을 실험자료를 통해 보았다.

지금까지의 실험을 통해 샘플링 주기를 줄이고, 신경세포의 수를 증가시키기 위해서는 신경회로망 전용 CPU의 도입이나, 병렬 처리구조의 제어기가 필요하였고, 보다 제어에 적절한 신경회로망의 구조와 입력항, 오차항의 정의에 관한 연구가 필요하며, 실제 현장에서 사용을 위한 수학적인 안정성 해석이 요구된다. 또한 신경회로망에 강인성을 부여하여, 학습에 많은 영향을 미치는 궤환 신호의 잡음 제거에 대한 연구가 필요하다.

참고문헌

[1] W. Thomas Miller III, Filson H. Glanz and L. Gordon Kraft III, "Application of a general learning algorithm to the control of robotic manipulators," *The International Journal of Robotics Research*, vol. 6, no. 2, pp. 84-98, Summer, 1987.

[2] Seul Jung and T. C. Hsia, "On reference trajectory modification approach for cartesian space neural network control of robot manipulators," IEEE International Conference on Robotics and

Automation, pp. 575-580, 1995.

[3] Seul Jung and T. C. Hsia, "On neural network application to robust impedance control of robot manipulators," *IEEE International Conference on Robotics and Automation*, pp. 869-874, 1995.

[4] Qing Li, A. N. Poo and C. L. Teo, "Disturbance rejection in the control of the robotic manipulators using a neural network structure," *ICARCV '92 Second International Conference on Automation*, Robotics and Computer Vision, vol. 3, RO-4.1.1~RO-4.1.5.

[5] Riko Šafarič and Karel jezernik, "Trajectory tracking neural network controller for a robot Mechanism and Lyapunov Theory of Stability," *Intelligent Robots and Systems*, vol. 1, pp. 626-633, Sep., 1994.

[6] Ahmad abdalla and Lilong Cai, "Stability analysis of robot manipulators subject to feedforward neural network controllers," *IEEE International Conference on Systems, Man*, and *Cybernetics*, vol. 3, pp. 2588-2593, Oct., 1994.

[7] P. M. Mills, A. Y. Zomaya and M. O. Tadė, "Adaptive model-based control using neural networks," *Int. J. Control*, vol. 60, no. 6, pp. 1163-1192, 1994.

[8] W. Thomas Miller, III, "Real-Time neural network control of a biped walking robot," *IEEE Control System*, pp. 41-48, Feb., 1994.

[9] D. C. Kar, K. Jayarajan and P. K. Pal, "Generation of optimal configuration for a redundant manipulator with a trained neural network," *Intelligent Robots and Systems*, vol. 3, pp. 2139-2145, 1994.

[10] K. Tsutumi and H. Matsumoto, "Neural computation and learning strategy for manipulator position control," *Proc. of the IEEE Int. Conf. on Neural Networks*, vol. 4, pp. 525-534, 1987.

[11] M. Kawato, H. Miyamoto, T. Setoyama, and R. Suzuki, "Feedback error learning neural network for trajectory control of a robotic manipulator," *Neural Networks*, vol. 1, pp. 251-265, 1988.

[12] 오 세영, 조 문중, 문 영주, "신경회로를 이용한 6축 로보트의 역동력학적 토크제어", 전기학회 논문지, 40권 8호, pp. 816~824, 1991.

[13] James A. Freeman, *Neural Networks Algorithms, Applications, and Programming Techniques*, Addison-Wesley, pp. 89-125, 1992.

[14] 이 상원, "학습하는 기계 신경망", Ohm사, pp. 266-285, 1993.

[15] Mohamad H. Hassoun, *Fundamentals of Artificial Neural Networks*, MIT Press, pp. 417-462, 1995.

[16] 강철구, 권인소, 윤중선, 정완균, "로봇 동역학과 제어", 희중당, pp. 177-180, 1994.

이 보 희

1985년 인하대 전공학과 졸업, 동대학원 석사(1992). 동대학원 기계공학과 박사(1996). 1997년 ~ 현재 세명대학교 전기공학과 교수. 관심분야는 이동로봇, 메카트로닉스.

김 진 걸

1978년 서울대 전기공학과 졸업, 1984년 IOWA(미)전기 및 컴퓨터공학과 석사. 1986년 IOWA(미) 수학과 석사. 1988년 IOWA(미) 전기 및 컴퓨터공학과 박사. 1988년 ~ 현재 인하대학교 자동화공학과 교수. 관심분야는 이동로봇, 최적제어, 비선형제어.

이 택 승

1994년 인하대 자동화공학과 졸업, 동대학원 석사(1996). 1996년 ~ 현재 LG 산전 연구소. 관심분야는 신경회로망, PLC.