

## 로봇-작업환경 동역학의 학습에 의한 로봇의 힘 추종 임피던스 제어

### Force Tracking Impedance Control of Robot by Learning of Robot-Environment Dynamics

신 상운\*, 최 규종\*, 이 규상\*, 김 영원\*, 안 두성\*\*

\* 부경대학교 기계공학과 대학원(tel:+82-051-620-6454)

\*\* 부경대학교 기계공학과(tel:+82-051-620-6454;FAX :+82-051-620-6454)

**Abstract** : Performance of force tracking impedance control of robot manipulators is degraded by the uncertainties in the robot and environment dynamic model. The purpose of this paper is to improve the controller robustness by applying neural network. Neural networks are designed to learn the uncertainties in robot and environment model for compensating the uncertainties. The proposed scheme is verified through the simulation of 2-DOF robot manipulator.

**Keywords** : Impedance control, Robot manipulator, Learning, Neural network.

#### 1. 서론

산업용 로봇을 현장의 여러 분야에서 보다 효율적으로 활용하기 위해서는 조립 작업, 디버링 등과 같이 작업 환경에 구속되어지는 작업(constrained space task)을 정확하게 수행할 수 있어야 한다. 이러한 구속 작업 시에는 매니플레이터가 작업 환경에 순응하기 위해서 접촉력이 조절되어야 하며, 이를 위해 지금까지 많은 연구가 행해졌다. 이러한 연구들은 크게 2가지 방법으로 나눌 수 있다. 즉, 첫 번째는 하이브리드 위치/힘 제어(hybrid position/force control) [1]로서 위치 제어와 힘 제어의 방향을 분리해서 각각 독립적으로 위치와 힘을 제어하는 방식이다. 그러나 이 방법은 작업 계획(task planning)과 제어 법칙 스위칭(control law switching) 등이 필요하고 비구속(unconstrained)과 구속(constrained)운동 사이의 천이(transition) 시의 불안정성을 일으키는 문제가 있다. 두 번째는 임피던스 제어(impedance control) [2]방식으로 말단부(end-effector)의 위치와 접촉력 사이의 관계를 임피던스로 보고, 이 임피던스를 유지하는 방법이다. 이 방법은 비구속 및 구속 운동을 동일하게 제어할 수 있고 비구속 운동과 구속 운동 사이의 안정적 천이 등 많은 장점을 가지고 있다. 결과적으로 임피던스 제어는 조립, 디버링 등의 작업에 적합한 것으로 인정되고 있다. 그러나 임피던스 제어에서의 접촉력의 크기는 말단부의 기준 위치 및 작업 환경의 강성(stiffness)과 위치(location)에 좌우되나 로봇과 작업 환경 동역학의 불확실성 때문에 정확한 접촉력 제어가 힘들다.

본 논문에서는 로봇 시스템에 중요하게 영향을 끼치는 로봇과 작업 환경에 대한 각종 불확실성이 있음에도 불구하고 힘 추종 임피던스 제어를 효율적으로 수행할 수 있는 신경회로망을 이용한 제어 기법을 제안한다.

#### 2. 로봇과 작업환경의 동역학 모델

매니플레이터가 환경과 접촉을 하면, 접촉력  $F_e$ 가 환경에 의해 말단부에 생기게 된다. 이러한 경우에 로봇의 운동방정식은 다음과 같다.

$$D^* \ddot{x} + h^* + f_j^* = F - F_e \quad (1)$$

여기서,  $D^*$ 는  $n \times n$  작업공간(Cartesian space) 관성 행렬,  $h^*$ 는 카테이시언 구심력과 Coriolis효과를 포함하는  $n \times 1$ 행렬,  $f_j^*$ 는 마찰력을 나타내는  $n \times 1$ 행렬,  $F$ 는  $n \times 1$ 의 제어입력인 힘,  $F_e$ 는  $n \times 1$ 의 접촉력,  $x$ 는  $n \times 1$ 의 말단부 위치이다.

실제로 위의 모델식에서  $D^*$ ,  $h^*$ 와  $f_j^*$ 는 정확하게 알 수가 없다. 따라서 단지 추정치를 사용하여 제어 설계를 한다.

작업 환경을 선형 스프링으로 모델하면 접촉력은 다음 식으로 표현된다.

$$F = k_e(x - x_e) \quad (2)$$

여기서,  $k_e$ 는  $n \times n$ 의 환경 강성계수 행렬이고,  $x$ 는 말단부의 위치이며,  $x_e$ 는 환경부위의 위치이다.

### 3. 임피던스 제어의 힘 추종 특성

임피던스 제어는 설계자가 지정하는 목표 임피던스(target impedance)를 말단부 위치와 접촉력 사이에서 유지시키는 것이다. 힘 추종(force tracking)을 위해서 목표 임피던스는 다음과 같이 된다.

$$M \frac{d^2}{dt^2} [x(t) - x_r(t)] + B \frac{d}{dt} [x(t) - x_r(t)] + K[x(t) - x_r(t)] = E_f(t) \quad (3)$$

여기서,  $E_f = F_r - F$ 이며,  $F_r$ 은 목표 접촉력,  $F$ 는 접촉력,  $x_r$ 은 기준위치 궤적,  $x$ 는 말단부의 실제 위치이다.

정확한 힘 추종을 위해서 기준 궤적을 다음과 같이 주면 된다 [3].

$$x_r = x_e + \frac{F_r}{k_e} \quad (4)$$

그러나  $x_e$ 와  $k_e$ 는 작업 환경의 불확실성 때문에 실제로 다음과 같게 된다.

$$x_r = \hat{x}_e + \frac{F_r}{\hat{k}_e} \quad (5)$$

로봇 매니플레이터의 위치 제어가 완전하다고 가정하면, 접촉력은 다음 식과 같이 된다.

$$F = F_r + \frac{Ms^2 + Bs + K}{Ms^2 + Bs + K + k_e} [k_e(\hat{x}_e - x_e) + \Delta k_e \frac{F_r}{\hat{k}_e}] \quad (6)$$

이 식의 두 번째 항은 작업 환경의 불확실성에서 기인하는 힘 추종 오차이다.

### 4. 신경회로망을 이용한 불확실성의 학습

본 장에서는 신경회로망을 이용하여 불확실성을 학습하여 힘 추종 오차를 제거시키고자 한다. 즉, 기준 위치  $x_r$ 는 다음과 같이 정의된다[4].

$$x_r = \hat{x}_e + \frac{F_r + F_{NN}}{\hat{k}_e} \quad (7)$$

여기서  $F_{NN}$ 은 신경회로망의 출력이다. 그림 1은 신경회로망에 기초를 둔 힘 추종 임피던스의 전체 구조이다. 신경회로망 I은 작업환경의 불확실성을 신경회로망 II는 로봇의 불활성성을 각각 보상하게 된다. 위치 제어 로봇이 완전하다면  $x_c = x$ 가 되며 다음의 식이 성립된다.

$$\begin{aligned} x_c &= x_r + \frac{1}{Ms^2 + Bs + K} (F_r - F) \\ &= \hat{x}_e + \frac{F_r + F_{NN}}{\hat{k}_e} + \frac{1}{Ms^2 + Bs + K} (F_r - F) \end{aligned} \quad (8)$$

이때의 접촉력은

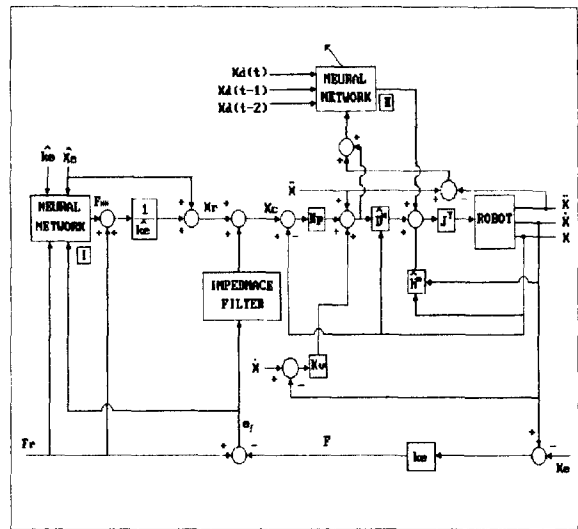


그림1. 신경회로망을 이용한 힘 추종 임피던스의 구조  
Fig. 1 Structure of force tracking impedance control using neural network

$$\begin{aligned} F &= k_e [\hat{x}_e + \frac{F_r + F_{NN}}{\hat{k}_e} + \frac{1}{Ms^2 + Bs + K} (F_r - F) - x_e] \\ &= F_r + k_e (\hat{x}_e - x_e) + \Delta k_e \frac{F_r}{\hat{k}_e} + k_e \frac{F_{NN}}{\hat{k}_e} \\ &\quad + \frac{k_e}{Ms^2 + Bs + K} (F_r - F) \end{aligned} \quad (9)$$

이 식을 정리하면

$$\begin{aligned} F_r - F &= \frac{Ms^2 + Bs + K}{Ms^2 + Bs + K + k_e} [k_e (\hat{x}_e - x_e) \\ &\quad + \Delta k_e \frac{F_r}{\hat{k}_e} + k_e \frac{F_{NN}}{\hat{k}_e}] \end{aligned} \quad (10)$$

[ ] 안의 항을 상수 또는 거의 상수로써 가정하고 정리하면

$$\begin{aligned} m \ddot{e}_f + b \dot{e}_f + k e_f &= \frac{k \hat{k}_e}{\hat{k}_e} [\hat{k}_e \Delta x_e + \frac{\Delta k_e}{k_e} F_r \\ &\quad - \frac{\hat{k}_e}{k} e_f + F_{NN}] \end{aligned} \quad (11)$$

여기서 에러신호  $v$ 를 다음과 같이 정의한다.

$$v = m \ddot{e}_f + b \dot{e}_f + k e_f \quad (12)$$

그러면 신경회로망의 목적은  $v = 0$ 이 되도록  $F_{NN}$ 을 출력하는 것이 된다.

환경의 불확실성에 대한 보상기로서 사용한 신경회로망은 2층으로 구성되고, 입력  $[\hat{x}_e \quad \hat{k}_e \quad F_d \quad e_f]$ 은  $w$ (weight)에 의해서 곱해지고, 각 은닉층의 노드에 합해진다. 은닉층에서는 사용되는 비선형 시그모이드 함수(sigmoid function)인  $f(\cdot)$ 를 사용하고, 출력층에서는 선형함수를 사용한다.

$$f(\cdot) = \frac{1}{1 + \exp(-(\cdot))} \quad (13)$$

그림2는 환경의 불확실성에 대한 보상기로서의 사용된 신경회로망의 구조이다.

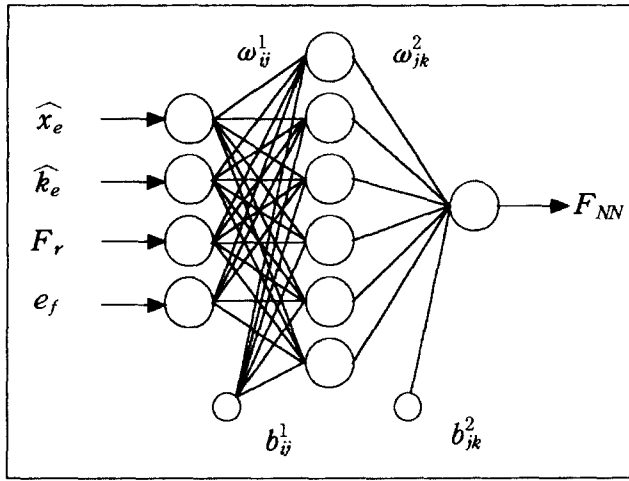


그림2 신경회로망의 구조  
Fig. 2 Structure of neural network

신경회로망은 다음의 비용함수(cost function), E를 최소화하면 된다.

$$E = \frac{1}{2} v^T v \quad (14)$$

위의 비용함수로부터 신경회로망의 학습법칙을 유도하기 위하여 신경회로망 내의 임의의 한 연결강도(weight),  $\omega$ 에 대한 그레디언트(gradient)를 구하면 아래와 같다.

$$\frac{\partial E}{\partial \omega} = \frac{\partial v^T}{\partial \omega} v = - \frac{\partial F_{NN}}{\partial \omega} v \quad (15)$$

여기서,  $\frac{\partial v^T}{\partial \omega} = - \frac{\partial F_{NN}}{\partial \omega}$  을 이용한다.

따라서 신경회로망의 학습법칙은 최종적으로 아래와 같이 구해진다.

$$\Delta \omega(t) = \eta \frac{\partial v^T}{\partial \omega} v + \alpha \Delta \omega(t-1) \quad (16)$$

여기서,  $\eta$  은 0보다 큰 상수로서 학습률을 나타내고,  $\alpha$  는 모멘텀 상수이다.

$$\Delta \omega_{ij}^1(t) = \eta y_j^1 (1 - y_j^1) x_i [ \sum_{k=1}^2 v_k \omega_{jk}^2 ] + \alpha \Delta \omega_{ij}^1(t-1) \quad (17)$$

$$\Delta \omega_{jk}^2(t) = \eta v_k y_j^1 + \alpha \Delta \omega_{jk}^2(t-1) \quad (18)$$

$$\Delta b_{ij}^1(t) = \eta y_j^1 (1 - y_j^1) [ \sum_{k=1}^2 v_k \omega_{jk}^2 ] + \alpha \Delta b_{ij}^1(t-1) \quad (19)$$

$$\Delta b_{jk}^2(t) = \eta v_k + \alpha \Delta b_{jk}^2(t-1) \quad (20)$$

$$y_j^1 = \frac{1}{1 + \exp(-(\sum_{i=1}^4 x_i \omega_{ij}^1 + b_j^1))} \quad (21)$$

$$y_k^2 = \sum_{j=1}^4 y_j^1 \omega_{jk}^2 b_k^2 \quad (22)$$

여기서,  $y_j^1$  은 은닉층의 j번째 출력이고,  $y_k^2$  는 출력층의 k번째 출력이다. 매니플레이터의 불확실성에 대한 학습은 [5]에 제시된 방법으로 실시한다.

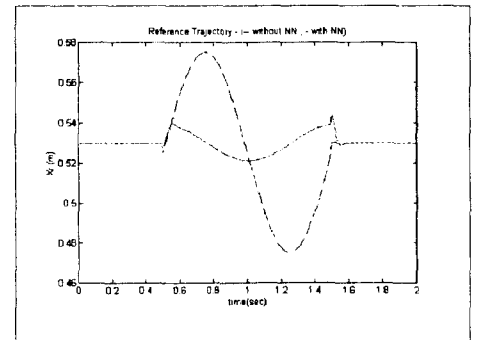
## 5. 시뮬레이션

시뮬레이션은 2축 로봇 매니플레이터를 사용하였다. 다음 표1은 로봇 매니플레이터를 나타낸다.

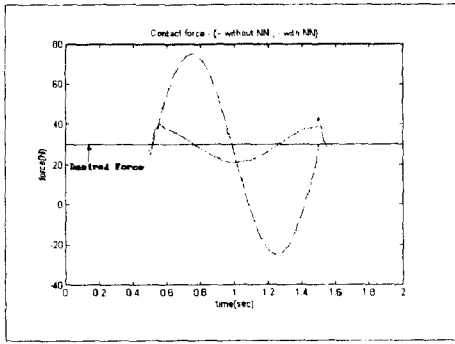
표1 로봇구성  
TABLE1 Parameter of manipulator

	Link1	Link2	Unit
Length	0.5	0.5	m
Mass	5	4.5	kg
Joint type	Revolute	Revolute	
C.of center	0.25	0.3	m

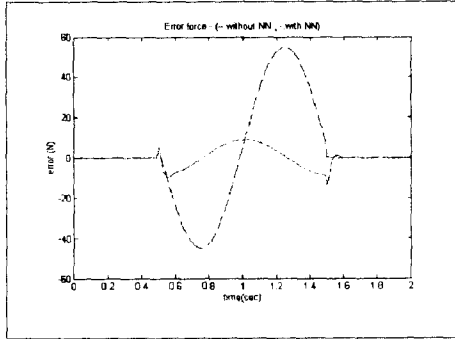
실험은 2가지 방법으로 실행하였다. 첫 번째는, 매니플레이터가 불확실성이 없고 완전한 위치 제어를 한다고 하지만, 환경의 불확실성이 존재하는 경우이다. 환경의 위치와 강성계수의 변화는  $x_e = 0.5 (0 \leq t < 0.5)$ ,  $x_e = 0.5 + 0.05 \sin(2\pi t) (0.5 \leq t \leq 1.5)$ ,  $x_e = 0.5 (1.5 < t \leq 2)$  고,  $\hat{k}_e$  는 1000(N/m)에서  $x_e = 0.5$ 의 지점에서 1200(N/m)으로 변화가 되고 기준힘은 30(N)으로 하여 실험을 하였다. 각각의 그래프는 신경회로망 보상기가 있는 경우와 없는 경우를 비교하였다. 그림3(a)은 기준 궤적에 관한 것이고, 그림3(b)는 힘 추종에 관한 것이고, 그림3(c)는 힘 추종 에러에 대한 그래프이다. 두 번째 실험은 로봇 매니플레이터의 불확실성과 환경의 불확실성이 둘 다 존재하는 경우에 대한 실험이다. 이번 실험에서는 환경의 불확실성에 대한 신경회로망 보상기와 로봇의 불확실성에 대한 신경회로망 보상기를 사용하여 로봇과 환경에 대해 불확실성을 보상해 주었다. 환경은  $x_e = 0.5 (0 \leq t < 1)$ ,  $x_e = 0.5 + 0.01 e^t (1 \leq t \leq 2)$ 로 변화가 되고, 강성계수는 1000(N/m)에서  $t=1(s)$ 인 지점에서 130(N/m)으로 변화가 되며, 기준힘은 30(N)으로 하고 실험을 하였다. 그림4(a)는 기준 궤적에 대한 것이고, 그림4(b)는 힘 추종에 관한 것이고, 그림4(c)는 힘 에러에 대한 그래프이다. 로봇 매니플레이터의 불확실성이 있는 경우에는 목표치에 대한 출력 값에 대한 에러가 생기기 때문에 기준 궤적 생성에 대한 신경회로망의 입력 값이 달라지므로 학습시에 약간의 오차가 더 나고 있지만 잘 추종한다는 것을 볼 수가 있다.



(a) 기준궤적 ( $x$ )

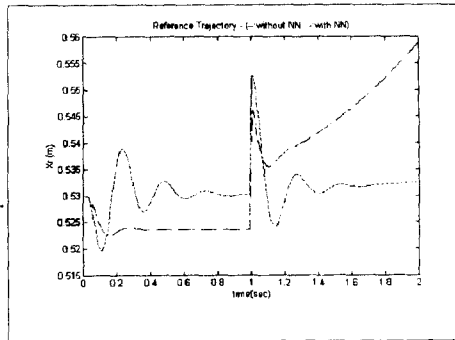


(b) 힘 추종 그래프

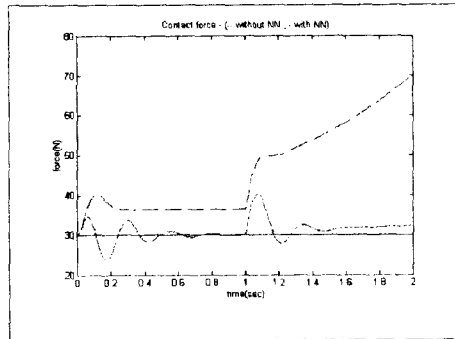


(c) 힘 추종에 대한 에러

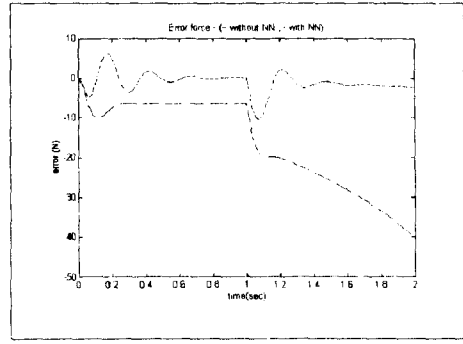
그림3 환경의 불확실성만 있는 경우의 결과  
Fig.3 Result of Environment uncertainly



(a) 기준궤적



(b) 힘 추종 그래프



(c) 힘 추종에 대한 에러

그림4 로봇의 불확실성과 환경의 불확실성이 있는 경우에 대한 결과

Fig.4 Result of Robot uncertainly and Environment uncertainly

## 6. 결론

본 논문에서는 로봇 불확실성과 작업환경에서의 불확실성이 있을 때 이를 극복할 수 있는 방법으로 신경회로망의 학습에 기초한 힘 추종 임피던스 제어기법을 소개하였다. 특히 작업환경의 불확실성을 극복하기 위해 제안된 학습법칙은 힘에 대한 오차 방정식에서 부터 쉽게 유도된다. 시뮬레이션 결과는 본 논문에서 제시된 방법이 로봇과 작업 환경에서의 불확실성을 학습하여 힘 추종 임피던스 제어를 효과적으로 수행할 수 있음을 보여주고 있다.

## 참고문헌

- [1] M.H. Raibert, and J.J. Craig, "Hybrid position/force control of manipulators." *ASME Journal Dynamics Systems, Measurement and Control*, 103(2), pp. 126-133, 1981.
- [2] Hogan, N. "Impedance Control", an approach to manipulation, part I-III. *ASME Journal of Dynamics System* 107(1) pp.1-24,1985
- [3] H. Seraji and R. Colbaugh, "Force Tracking in Impedance Control", *Journal of Robotics Research*, Vol. 16, No 1. pp. 97-117. 1997
- [4] S. Uran, A. Bace, K. Jezernik, "Neural Network based Impedance Controller", 5-th. *Int. Workshop on Robotics in Alpe-Adria-Danube Region*.
- [5] S. Jung and T.C. Hsia, "On Neural Network Application to Robust Impedance Control of Robot Manipulators", *IEEE International Conference of Robotics and Automation*, pp. 869-874, 1995.