RBF 신경망을 이용한 로봇 매니퓰레이터의 분산제어

Decentralized Control of Robot Manipulator Using the RBF Neural Network

원 성 운*, 김 영 태**

* 동국대학교 전기공학과(전화:(011)9408-0032, E-mail : starcloud76@dreamwiz.com)
** 동국대학교 전기공학과(전화:(02)2260-3349, E-mail : youngtae@dongguk.edu)

Abstract : Control of multi-link robot arms is a very difficult problem because of the highly nonlinear dynamics. Decentralized control scheme is developed for control of robot manipulators based on RBF(Radial Basis Function) Neural Networks. RBF Neural Networks is used to approximate the coupling forces among the joints, coriolis force, centrifugal force, gravitational force, and frictional force. The compensation controller is also proposed to estimate the bound of approximation error so that the chattering effect of the control effort can be reduced. The proposed scheme does not require an accurate manipulator dynamic, and it is proved that closed-loop system is asymptotic stable despite the gross robot parameter variations. Numerical simulations for two-link robot manipulator are included to show the effectiveness of controller.

Keywords : Neural networks, Robot Control, Decentralized Control, Radial Basis Function, Adaptive Control

Ⅰ. 서론

로봇 매니퓰레이터는 여러 개의 관절이 연결되어 있어서 각 축의 움직임은 서로 근접한 다른 축에 영향을 미치게 되며 구조적으로도 특성이 매우 복잡하고 비선형적이다. 따라서 로봇의 운동을 정확하게 제어하기 위해서는 비선형 제어기가 필요하며 특히 경우에 따라서 외란과 미지의 부하나 마찰력이 발생하고 로봇이 빠르게 움직일 경우에 모델의 불확실성이 증가되므로 고도의 기능을 가진 제어 기법이 요구된다.

이러한 문제를 해결하기 위해 본 논문에서는 계산시간을 줄일 수 있고 실시간 제어가 용이하며 제어기를 간단하게 구현할 수 있어서 대형 시스템 제어에 많이 쓰이는 분산제어 이론과 RBF 신경망 이론을 조합한 제어방법을 제시하였다. RBF 신경망은 관절 사이의 연결력, 코리올리스(coriolis)힘, 원심력, 중력, 그리고 마찰력을 근사화 하는데 사용되었기 때문에 논문에서 제안한 제어이론은 로봇 매니퓰레이터의 정확한 동적 모델을 요구하지 않으며 여기에 보상 제어기를 추가하여 근사오차를 추정하고 채터링(chattering) 감소 효과를 얻을 수 있었다. 또한 정상 상태에서의 오차를 줄이기 위해 여과된 추적오차(filtered tracking error)에 적분항을 포함 시켰으며 궤환 미분항(feedback derivative term)은 Lyapunov 함수의 일차 미분값이 확실히 음의 부호를 갖게 하기 위하여 고려하였다. 또한 모의실험을 통해 제안한 방법을 2관절 로봇 매니퓰레이터에 적용시켜 그 효율성을 입증하였다.

Ⅱ. RBF 신경망

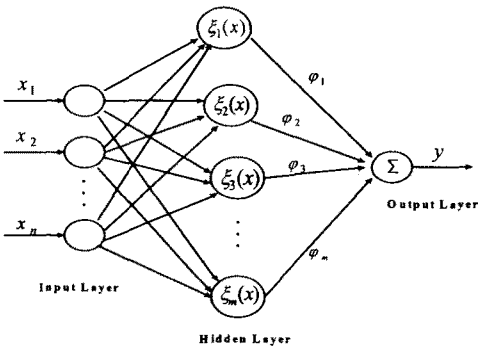

그림 1. RBF 신경망
Fig. 1. RBF Neural Network

그림 1과 같은 RBF 신경망은 Moody, Darken[1]에 의해 제안되었으며 어떤 시스템의 불확실성에 대해 적응적으로 학습하고 신경망의 출력은 시스템의 불확실성을 보상하기 위한 제어기의 변수로 사용된다. 또한 이 방법을 사용하면 불확실한 모델과 비선형성에 대해 강인함을 얻을 수 있을 뿐만 아니라 플랜트의 출력과 기준 값 사이의 추적오차(tracking error)가 점근적으로 0에 가깝게 된다[2]. 기본적인 RBF 신경망은 그림 1과 같이 입력층(input layer), 은닉층(hidden layer), 출력층(output layer)으로 구성되어 있으며 은닉층의 노드수는 사용자에 의해서 결정된다. 은닉층과 출력층 사이에는 서로를 연결해 주는 가중치가 존재하며 입력 벡터 $\underline{x} = (x_1, x_2, \cdots, x_n)^T \in X \subset R^n$ 을 출력 값 $y \in Y \subset R$에 대응시키는 비선형 함수의 역할을 한다. 은닉층에 사용되는 함수로는 일반적으로 Gaussian 함수를 많이 사용하며 본 논문에서도 은닉층 함수로

Gaussian 함수를 사용하였다. 또한 RBF 신경망의 출력은 가중치의 합(weighted sum)을 이용하여 구하였으며 Gaussian 함수를 이용하여 i번째 노드의 출력을 나타내면 다음과 같다.

$$\xi_i(\underline{x}) = exp\left(-\frac{\|\underline{x}-c_i\|^2}{\sigma_i^2}\right) \quad (1)$$

여기서, c_i는 Gaussian함수의 중심값이고, σ_i는 Gaussian 함수의 폭이다. RBF 신경망의 전체출력은 다음과 같이 각 노드의 가중치 합으로 표현된다.

$$y(\underline{x}) = \sum_{i=1}^{m} \varphi \xi_i(\underline{x}) = \varphi^T \xi(\underline{x}) \quad (2)$$

여기서, $\varphi = (\varphi_1, \varphi_2, \cdots, \varphi_m)^T$은 가중치 벡터이고, $\xi(\underline{x}) = (\xi_1(\underline{x}), \xi_2(\underline{x}), \cdots, \xi_m(\underline{x}))^T$은 Gaussian함수의 벡터이다. 식(2)와 같은 RBF 신경망은 보편적 근사기임을 알 수 있으며[3] 콤팩트(compact) 입력 공간에서 어떠한 정도의 정확성에 대해서 모든 비선형 함수를 근사화 할 수 있다.

III. RBF 신경망 분산제어기 설계

일반적인 로봇 매니퓰레이터의 Lagrange-Euler 운동 방정식을 행렬 벡터 표기로 나타내면 다음과 같다.

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) + H(\dot{\theta}) = \tau(t) \quad (3)$$

여기서, $\theta(t)$는 관절각 위치 벡터, $\tau(t)$는 관절 토크 벡터, $G(\theta)$는 중력 벡터, $M(\theta)$는 대칭 양한정(symmetric positive-definite) 관성벡터, $C(\theta, \dot{\theta})$는 코리올리스와 원심력 벡터, 그리고 $H(\dot{\theta})$는 마찰력 벡터이다. 다관절 로봇 시스템은 각각의 관절이 결합력(coupling force)에 의해 상호 연결된 종속 시스템들로 구성된 대형 시스템으로 생각할 수 있다. 따라서 식(3)은 다음과 같은 n개의 2차 비선형 스칼라 미분 방정식으로 표현할 수 있다.

$$m_{ii}(\theta)\ddot{\theta}_i(t) + \sum_{\substack{j=1 \\ j\neq i}}^{n} m_{ij}(\theta)\ddot{\theta}_j(t) + c_i(\theta, \dot{\theta}) + g_i(\theta) + h_i(\dot{\theta}) = \tau_i$$

$$i = 1, 2, \cdots, n \quad (4)$$

여기서, 관성 행렬 $M(\theta)$는 양한정 이기 때문에 $m_{ii}(\theta)$는 항상 양수이며 i번째 관절에서의 시변 유효 관성으로 볼 수 있다. 결과적으로 식(3)은 다음과 같이 나타낼 수 있다.

$$m_{ii}(\theta)\ddot{\theta}_i(t) + d_i(\theta, \dot{\theta}, \ddot{\theta}) = \tau_i(t) \quad (5)$$

$$i = 1, 2, \cdots, n$$

여기서, $d_i(\theta, \dot{\theta}, \ddot{\theta}) = \sum_{\substack{j=1 \\ j\neq i}}^{n} m_{ij}(\theta)\ddot{\theta}_j(t) + c_i(\theta, \dot{\theta}) + g_i(\theta) + h_i(\dot{\theta})$

이다. 식(5)는 관절 토크 $\tau_i(t)$를 가진 i번째 종속 시스템의 입출력 동적 모델로서 생각 할 수 있으며 $d_i(\theta, \dot{\theta}, \ddot{\theta})$은 i번째 종속 시스템과 나머지 다른 종속 시스템들 사이의 결합(coupling)을 나타낸다. 따라서 관절 제어를 분산화 할 수 있으며 분산 매니퓰레이터 제어문제는 i번째 관절 토크 $\tau_i(t)$를 독립적으로 얻을 수 있는 관절 제어기를 설계하는 문제로 생각할 수 있다. 분산 제어기를 설계하기 위하여 식(6)과 같은 여과된 추적 오차(filtered tracking error)를 정의 하자.

$$e_i(t) = \dot{\theta}_i - \dot{\theta}_{r_i} \quad (6)$$

$$i = 1, 2, \cdots, n$$

여기서, $\dot{\theta}_{r_i} = \dot{\theta}_{d_i} - k_{i1}(\theta_i - \theta_{d_i}) - k_{i2}\int_0^t (\theta_i - \theta_{d_i})dt$ 이며, θ_{d_i}는 기준 변위각이다. 식(5)과 식(6)으로부터 식(7)을 얻을 수 있다.

$$\begin{aligned} m_{ii}\dot{e}_i(t) &= m_{ii}(\theta)\ddot{\theta}_i(t) - m_{ii}(\theta)\ddot{\theta}_{r_i}(t) - \triangle m_i\dot{e}_i \\ &= -d_i(\theta, \dot{\theta}, \ddot{\theta}) - m_{ii}(\theta)\ddot{\theta}_{r_i}(t) + \tau_i(t) - \triangle m_i\dot{e}_i \\ &= f_i + \tau_i(t) \end{aligned} \quad (7)$$

여기서, f_i는 비선형 함수이고, $m_{ii}(\theta)$는 항상 양수인 i번째 관절의 시변 유효 관성이므로 기준경로가 주어지면 항상 최소 한계값 m_{ii}가 존재한다. 즉

$$0 < m_{ii} < m_{ii}(\theta), \quad m_{ii}(\theta) = m_{ii}(\theta) + \triangle m_i(\theta)$$

만일 유효관성 $m_{ii}(\theta)$와 i번째 종속 시스템과 나머지 다른 종속 시스템 사이의 결합력을 나타내는 $d_i(\theta, \dot{\theta}, \ddot{\theta})$을 정확히 알 수 있다면 최적 제어기를 쉽게 설계할 수 있지만[4] 실제적인 문제에서는 f_i의 정확한 값을 얻기 어려우므로 **known function** f_{ki}와 시스템의 불확실성을 나타내는 **unknown function** f_{ui}의 합으로 볼 수 있다. 본 논문에서는 f_{ui}를 RBF 신경망 $\hat{f}_{ui}(\theta|\varphi_i)$로 대체하여 식(8)과 같은 i번째 관절에서의 RBF 신경망 제어기를 제안하자.

$$\tau_i(t) = -f_{ki} - \hat{f}_{ui}(\theta|\varphi_i) - k_{i3}e_i(t) - \tau_{ic}(t) \quad (8)$$

여기서, $k_{i3}e_i(t)$는 Lyapunov 함수의 미분 값 $\dot{v}_i$을 더 작은 음수로 만들기 위해 고려하였고, 보상 제어기 τ_{ic}는 $\hat{f}_{ui}(\theta|\varphi_i)$와 f_{ui}간의 최소 근사오차 w_i를 보상하기 위해 설계되었다. 식(8)을 식(7)에 대입하면

$$m_{ii}\dot{e}_i(t) = f_{ui} - \hat{f}_{ui}(\theta|\varphi_i) - k_{i3}e_i(t) - \tau_{ic}(t) \quad (9)$$

이 되고, RBF 신경망 제어기의 최적 변수 벡터 φ_i^*와 최소 근사오차 w_i를 각각 식(10)와 식(11)로 정의하자.

$$\varphi_i^* = arg\min_{\varphi_i}\left[\sup_{\theta \in V}\left|f_{ui} - \hat{f}_{ui}(\theta|\varphi_i)\right|\right] \quad (10)$$

$$w_i = f_{ui} - \hat{f}_{ui}(\theta|\varphi_i^*) \quad (11)$$

위의 두 식을 이용해서 식(9)를 다시 쓰면

$$m_{ii}\dot{e}(t) = -k_{i3}e_i(t) + \hat{f}_{ui}(\theta|\varphi_i^*) - \hat{f}_{ui}(\theta|\varphi_i) + w_i - \tau_{ic}(t) \quad (12)$$

$\hat{f}_{ui}$을 식(2)와 같은 RBF 신경망으로 대체하면 식(12)는

$$m_{ii}\dot{e}(t) = -k_{i3}e_i(t) + \phi_i^T\xi_i(\theta) + w_i - \tau_{ic}(t) \quad (13)$$

여기서, $\phi_i = \varphi_i^* - \varphi_i$이고 $\xi_i(\theta)$은 식(1)과 같은 가우

시안 함수이다.

i번째 관절의 안정도 해석을 위해 다음과 같은 Lyapunov 함수를 고려하자.

$$v_{i1} = \frac{1}{2} m_i e_i^2 + \frac{1}{2 r_i} \phi_i^T \phi_i \tag{14}$$

여기서, e_i는 식(6)이며 r_i는 양의 상수이다. 여과된 추적오차를 따라 v_{i1}을 미분하면

$$\dot{v}_{i1} = m_i e_i \dot{e}_i + \frac{1}{r_i} \phi_i^T \dot{\phi}_i$$

$$= e_i [- k_{i3} e_i + \phi_i^T \xi_i(\theta) + w_i - \tau_{ic}] + \frac{1}{r_i} \phi_i^T \dot{\phi}_i$$

$$= - k_{i3} e_i^2 + \frac{1}{r_i} \phi_i^T [r_i e \xi_i(\theta) + \dot{\phi}_i] + e_i w_i - e_i \tau_{ic} \tag{15}$$

보편적 근사정리[3]에 의해 w_i는 양의 상수 $w_{i\triangle}$에 의해 $|w_i| \le w_{i\triangle}$와 같이 한정되고 매우 작은 값이 됨을 알 수 있다. 만약 RBF 신경망 가중치 갱신(tuning)에 식(16)과 같은 적응법칙을 취하고 식(17)과 같은 보상 제어 법칙을 선택하면

$$\dot{\varphi}_i = - \dot{\phi}_i = r_i e \xi_i(\theta) \tag{16}$$

$$\tau_{ic} = w_{i\triangle} sgn(e_i(t)) \tag{17}$$

여기서, $sgn(\cdot)$은 sign함수이다. 따라서 식(15)는 다음과 같이 다시 쓸 수 있다.

$$\dot{v}_{i1} = - k_{i3} e_i^2 + e_i w_i - |e_i| w_{i\triangle}$$

$$\le - k_{i3} e_i^2 + |e_i| |w_i| - |e_i| w_{i\triangle}$$

$$= - k_{i3} e_i^2 - |e_i| (w_{i\triangle} - |w_i|)$$

$$\le 0 \tag{18}$$

Babalat의 정리[5]를 사용하면, e_i가 0으로 수렴함을 알 수 있고 따라서 i번째 종속 시스템은 점근적으로 안정하게 된다. 하지만 식(17)의 sign함수는 실제 응용에서 채터링(chattering)을 야기 시키므로, 보상 제어 법칙을 이용하기 위해서는 근사오차의 한정을 알아야 한다. 이 문제를 해결하기 위해 한정 추정법(bound estimation scheme)을 RBF 신경망과 연계하고 보상제어법칙에 따라 $w_{i\triangle}$을 $\hat{w}_{i\triangle}$으로 바꾸면 식(17)은 다음과 같이 다시 쓸 수 있다.

$$\tau_{ic} = \hat{w}_{i\triangle}(t) sgn(e(t)) \tag{19}$$

여기서, $\hat{w}_{i\triangle}$는 근사오차의 한정 추정값이다. 추정 오차(estimation error)를 다음과 같이 정의하고

$$\tilde{w}_{i\triangle}(t) = w_{i\triangle} - \hat{w}_{i\triangle}(t) \tag{20}$$

다음과 같은 Lyapunov함수 후보를 고려한다.

$$v_{i2} = \frac{1}{2} m_i e_i^2(t) + \frac{1}{2 r_{i1}} \phi_i^T \phi_i + \frac{1}{2 r_{i2}} \tilde{w}_{i\triangle}^2(t) \tag{21}$$

여기서, r_{i1}과 r_{i2}는 양의 상수이고, 여과된 추적오차를 따라 v_{i2}를 미분하면

$$\dot{v}_{i2} = m_i e_i \dot{e}_i + \frac{1}{r_{i1}} \phi_i^T \dot{\phi}_i + \frac{1}{r_{i2}} \tilde{w}_{i\triangle} \dot{\tilde{w}}_{i\triangle}$$

$$= e_i [- k_{i3} e_i + \phi_i^T \xi_i(\theta) + w_i - \hat{w}_{i\triangle}(t) sgn(e_i(t))] + \frac{1}{r_{i1}} \phi_i^T \dot{\phi}_i + \frac{1}{r_{i2}} \tilde{w}_{i\triangle} \dot{\tilde{w}}_{i\triangle}$$

$$= - k_{i3} e_i^2(t) + \frac{1}{r_{i1}} \phi_i^T [r_{i1} e \xi_i(\theta) + \dot{\phi}_i] + e_i w_i - w_{i\triangle} |e_i| + \frac{1}{r_{i2}} \tilde{w}_{i\triangle} [r_{i2} |e_i| + \dot{\tilde{w}}_{i\triangle}] \tag{22}$$

만약 RBF신경망 가중치 갱신에 식(23)와 같은 적응법칙을 취하고 식(24)와 같은 보상제어법칙을 선택하면

$$\dot{\varphi}_i = - \dot{\phi}_i = r_{i1} e \xi_i(\theta) \tag{23}$$

$$\dot{\hat{w}}_{i\triangle} = - \dot{\tilde{w}}_{i\triangle} = r_{i2} |e_i| \tag{24}$$

따라서 식(21)은 다음과 같이 다시 쓸 수 있다.

$$\dot{v}_{i2} = - k_{i3} e_i^2 + e_i w_i - |e_i|$$

$$\le - k_{i3} e_i^2 + |e_i| w_i - |e_i| w_{i\triangle}$$

$$= - k_{i3} e_i^2 - |e_i| (w_{i\triangle} - |w_i|)$$

$$\le 0 \tag{25}$$

식(25)를 통해 시스템이 한정추정을 갖는 제안된 RBF 신경망 분산 제어기로 전체적인 점근 안정임을 알 수 있다.

Ⅳ. 모의실험

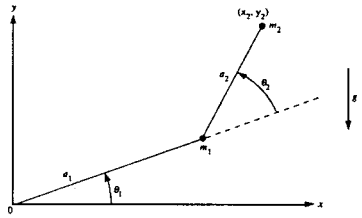

그림 2. 2관절 로봇 매니퓰레이터

Fig. 2. 2-link Robot Manipulator

3절에서 제안한 RBF신경망 분산제어기의 효율성을 보이기 위하여 그림 2와 같은 2관절 로봇 매니퓰레이터에 적용하였으며 동적 방정식[6]은 다음과 같다.

$$\begin{vmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{vmatrix} + \begin{vmatrix} c_1(\theta, \dot{\theta}) \\ c_2(\theta, \dot{\theta}) \end{vmatrix} + \begin{vmatrix} g_1(\theta) \\ g_2(\theta) \end{vmatrix} = \begin{vmatrix} \tau_1 \\ \tau_2 \end{vmatrix}$$

여기서, 파라메터 값들은 Lewis[7]등의 값을 사용하였으며 Gaussian함수의 중심값과 폭은 Zhihong[2]등의 값을 참고로 하였다.

- 기준궤적 경로

  $\theta_{d_1}(t) = 0.15 [0.8\pi t - sin(0.8\pi t)]$

  $\theta_{d_2}(t) = 0.15 [-0.8\pi t + sin(0.8\pi t)]$

- $m_1 = 0.8kg, \quad m_2 = 2.3kg, \quad l_1 = l_2 = 1m$
- gravity=9.8m/s^2
- $r_1 = 30000, \; r_2 = 30000$
- $k_{11} = k_{21} = 10, \quad k_{12} = k_{22} = 80, \quad k_{13} = k_{23} = 500$

그림 3는 RBF신경망 분산제어기를 적용한 시스템의 블록 다이어 그램이고, 그림 4과 그림 5는 초기위치가 각각 $(0.0)^T$과 $(0.3,-0.3)^T$일때 관절각의 시간에 대한 응답을 나타낸 것이며 그림 6와 그림 7은 초기위치$(0,0)^T$에 대해서 한정추정이 있을 때와 없을 때의 관절 토크의 시간에 대한 응답을 나타낸 것이다.

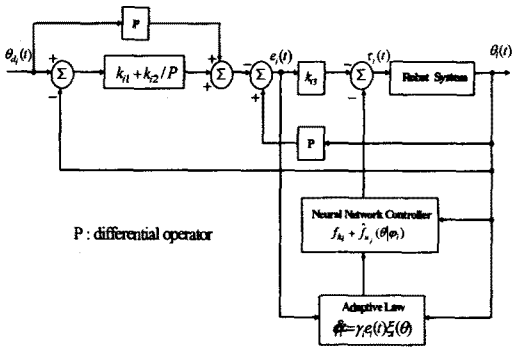

그림 3. RBF 신경망 분산 제어기를 갖는 로봇 시스템
Fig. 3. Robot System with RBF Neural Network Decentralized controller

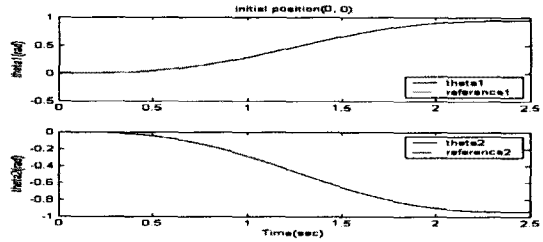

그림 4. 초기위치(0,0)에 대한 관절각의 궤적
Fig. 4. Trajectories of the joint angles for initial position(0, 0)

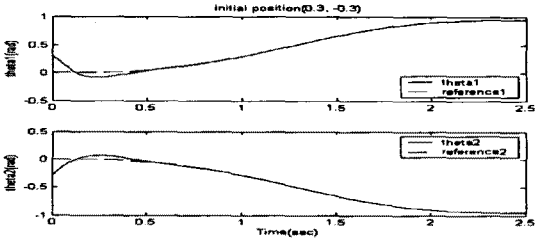

그림 5. 초기위치(0.3,-0.3)에 대한 관절각의 궤적
Fig. 5. Trajectories of the joint angles for initial position(0.3,-0.3)

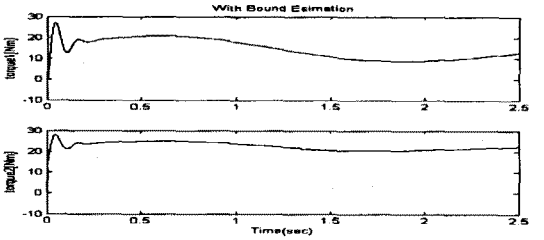

그림 6. 한정 추정이 있을 때의 관절토크의 궤적
Fig, 6. Trajectories of the joint torque with bound estimation

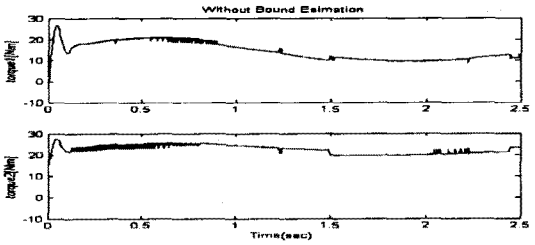

그림 7. 한정 추정이 없을 때의 관절토크의 궤적
Fig. 7. Trajectories of the joint torque without bound estimation

V. 결론

본 논문에서는 RBF신경망을 이용하여 로봇 매니퓰레이터의 RBF신경망 분산제어기를 제안하였다. RBF신경망을 이용해 알 수 없는 비선형 함수를 근사화 할 수 있었고 또한 분산제어 이론을 적용하여 실시간 제어하는데 있어서 시간을 많이 줄일 수 있었으며 제어기를 간단하게 구현할 수 있음을 보였다. 여과된 추적오차에 적분항을 포함시켜 정상상태 오차를 줄일 수 있었고, 보상 제어기를 추가하여 근사오차 추정과 채터링 감소효과를 얻을 수 있었다. 또한 제안된 제어기는 Lyapunov 함수의 일차 미분 값이 확실히 음의 부호를 갖게 하여 결과적으로 점근 안정임을 알 수 있었으며 모의실험을 통해 논문에서 제안한 제어기의 효율성을 입증할 수 있었다.

참고문헌

[1] J. Moody and C. Darken, "Fast learning in networks of locally-tuned processing units," Neural Computation, vol. 1, pp. 281-294, 1989.

[2] M. Zhihong, H. R. Wu, and M. Palaniswame, "An adaptive tracking controller using neural networks for a class of nonlinear systems," IEEE Trans. Neural Networks, vol. 9, no. 5,pp. 947-955, Sep. 1998.

[3] J. Park and J. W. Sandberg, "Universal approximation using radial basis function networks," Neural Computation, vol. 3, pp. 246-257, 1991.

[4] S. Sastry and M. Bodson, "Adaptive Control : Stability, Convergence, and Robustness," Prentice Hall, 1989.

[5] J. J. E. Slotine and W. Li, "Applied Nonlinear Control," Prentice Hall, 1991.

[6] F. L. Lewis, C. T. Abdallah, and D. W. Dawson, "Control of Robot Manipulator," New York: Macmillan, 1993.

[7] F. L. Lewis, A. Yesildrek, K. Liu, "Multilayer Neural Net Robot Controller with Guaranteed Tracking Performance," IEEE Trans. Neural Networks, vol. 7, no. 2, pp. 388-399, 1996.