

論文96-33B-5-4

퍼지-신경망 제어기를 이용한 불확실한 로봇 매니플레이터의 적응 학습 제어

(Adaptive Learning Control of an Uncertain Robot Manipulator Using Fuzzy-Neural Network Controller)

金成炫*, 崔永吉*, 金容虎*, 全洪兌*

(Seong Hyun Kim, Young Keel Choi, Yong Ho Kim, and Hong Tae Jeon)

요약

최근까지 퍼지 및 신경망 제어기를 이용하여 비선형 시스템을 제어하기 위해 많은 연구가 되었다. 본 논문은 이러한 분류의 연구로서 비선형 시스템인 로봇 매니플레이터를 제어하기 위해 적응 제어 방식과 지능제어 이론을 도입한 직접 적응 학습 제어 방식을 제안한다. 제안된 방식은 로봇의 온라인 제어에 적용된다. 이때 로봇 매니플레이터의 비선형 동적특성은 불확실한 것으로 가정되어 진다. 제안된 방식의 효율성은 2 자유도 매니플레이터를 사용하여 모의 실험을 통하여 검토된다.

Abstract

This paper will propose the direct adaptive learning control scheme based on adaptive control technique and intelligent control theory for a nonlinear system. Using the proposed learning control scheme, we will apply to on-line control an uncertain robot manipulator. It is assumed that the dynamic characteristic of a robot manipulator is uncertain but for model perfect matching, its structure condition is known. The effectiveness of the proposed control scheme will be illustrated by simulations of a robot manipulator.

I. 서론

최근까지 비선형 시스템을 효율적으로 제어하기 위해서 퍼지와 신경망 제어이론과 같은 지능 제어이론이 많이 연구되었다. 일반적으로 지능 제어이란 불확실한 제어환경과 상호 작용하면서 제어목적 성취하기 위해 스스로 조직화하고 학습할 수 있는 적응 제어를 말한다. 즉, 지능 제어기법은 기존의 수학적 분석이 용이하지 못한 제어 응용에 적합한 기법이다. 이러한 지능 제어를 실현하기 위한 대표적인 병렬 접근 방식으로 퍼지 논리(fuzzy logic)^[1]와 신경망(neural net-

works)^[2] 방식들이 있다. 그리고 두 접근 방식의 보상적 특성 및 합성, 결합에 대한 연구도 시도되었다^[3]. 퍼지 및 신경망을 포함한 지능 제어시스템은 크게 두 가지 제어방식 즉 간접, 직접 방식으로 분류된다. 간접 방식에 해당하는 대표적인 연구결과로는 Jordan^[4], Narendra^[5]가 제시한 신경망 제어시스템으로 플랜트의 식별기(identifier)와 제어기에 해당하는 두개의 신경망으로 구성된다. 그리고 Hasegawa^[6]은 식별기와 제어기를 퍼지 신경망으로 구성하여 학습속도 및 제어 성능을 향상하도록 제안하였다. 이러한 방식은 플랜트의 식별기와 제어기의 분리된 학습과정을 필요로 하고 모델로부터 역전파된 오차량으로 제어기를 학습한다. 그러므로 식별기의 학습정도에 따라 제어기의 성능이 좌우되는 특성이 있다. 또한 플랜트의 매개변수들이 변

* 正會員, 中央大學校 電子工學科

(Dept. of Elec. Eng., Chung-Ang Univ.)

接受日字:1995年10月27日, 수정완료일:1996年4月23日

할 경우 식별기의 재학습이 요구된다. 따라서 식별기를 필요로 하지 않는 직접 제어방식이 간접 제어방식보다 온라인 제어방식에 적합하며 지능 제어기의 장점을 최대한 활용할 수 있다. 직접 제어방식을 적용하기 위해서는 플랜트의 동적특성을 직접 고려하여야 한다. 그러나 플랜트의 동적특성이 불확실할 경우 Yabuta^[7]가 논의한 바와 같이 플랜트의 출력으로 부터 제어기의 학습정보를 얻기 힘들다. 이러한 문제점을 해결할 수 있는 방식으로 Barto^[8]는 강화학습(reinforcement learning)을 갖는 신경망 제어시스템을 제시하였다. Berenji^[9]는 이 결과를 퍼지-신경망으로 확장하여 빠른 학습속도를 갖도록 유도하였다. 상기의 강화학습에 의한 방식은 실패여부를 예측하여 제어기의 성능을 향상시키는 방식으로 온라인 학습 과정에서 안정한 제어를 보장하기 힘들다.

따라서 본 논문에서는 불확실한 로봇트의 온라인 학습 제어를 위해서 식별기가 필요치 않는 직접 제어 방식을 채택한다. 그리고 퍼지 신경망을 이용하여 온라인 학습 제어시 안정성을 보장하도록 Lyapunov 안정성 이론을 바탕으로 적응 학습 규칙을 유도한다. 또한 로봇트의 불확실한 동적특성에 대한 퍼지 신경망제어기의 강건성에 대해 검토된다.

II. 지능 제어를 위한 퍼지-신경망

제어대상과 실행지표 혹은 제어목적이 주어졌을 때 제어 규칙 $f^*: X \rightarrow U$ 가 존재한다고 가정하자. 만약 제어 대상이 비선형 시스템일때 제어 규칙 f^* 는 복잡한 비선형 함수가 될 수 있다. 그러므로 지능 제어기는 이러한 비선형 함수를 잘 표현할 수 있어야 한다. 이러한 특성은 비선형 함수 f^* 의 근사화 능력에 의해 좌우된다.

퍼지 시스템과 신경망은 각각 비선형 처리 소자인 지지 함수(support function)에 의해서 입출력 비선형 맵핑 $f: X \rightarrow U$ 을 수행한다. 퍼지 시스템에서 지지함수는 언어적 규칙을 표현할 수 있는 임의의 함수가 될 수 있다. 그리고 신경망에서 지지 함수는 방사형 함수(radial function), 스플라인(splines), 다항 함수(polynomial function), 그리고 시그모이드 함수(sigmoid functions)가 될 수 있다. 일반적으로 퍼지와 신경망은 비선형 지지함수를 이용하여 임의의 비선형 함수를 ϵ 의 범위에서 근사화할 수 있는 것으로 증명되었다^[10]

^[11]. 이 속성은 퍼지와 신경망이 비선형 제어를 수행하는 이론적 근거를 마련해 준다. 본 절에서는 제어 규칙 f^* 를 잘 표현할 수 있는 퍼지 방사형 함수망(Fuzzy Radial Function Network)에 대해서 서술된다.

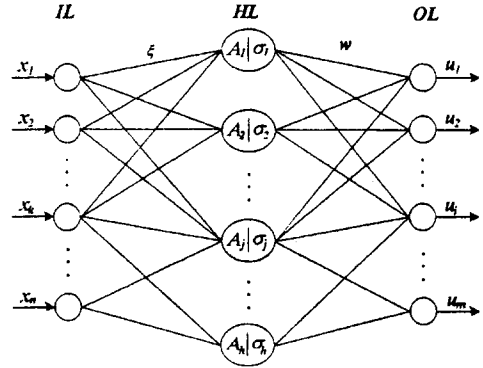


그림 1. 퍼지 방사형 함수망
Fig. 1. Fuzzy radial function network.

1. 퍼지 방사형 함수망

그림 1은 입력층(IL), 중간층(HL), 그리고 출력층(OL)으로 구성된 퍼지 방사형 함수망을 보인다. 중간층의 출력 $s_j(x)$ 는 가우시안 함수를 사용하여 퍼지 규칙의 소속정도를 다음과 같이 계산한다.

$$s_j(x) = \exp \left[- \frac{(x - \xi_j)^T A_j (x - \xi_j)}{\sigma_j^2} \right] \quad (1)$$

여기서, $j=1,2,\dots,h$ 이고 h 은 중간층의 유니트 수이다. 그리고 $x = [x_1, x_2, \dots, x_n]^T$, ξ_j , σ_j 은 각각 가우시안 함수의 입력, 중심 벡터 및 폭에 해당된다. 또한 $A_j = \text{diag}(a_{j1}, a_{j2}, \dots, a_{jn})$ 는 가중행렬이다.

출력층의 출력 u_i 는 중간층의 출력 s_j 로부터 다음과 같이 가중 평균화된 제어량을 계산한다.

$$u_i = \sum_{j=1}^h w_{ij} s_j / \sum_{j=1}^h s_j = \sum_{j=1}^h w_{ij} \phi_j \quad (2)$$

여기서, w_{ij} 는 출력층의 i 번째 유니트와 HL의 j 번째 사이의 가중치이고, $i=1,2,\dots,m$, 이며 m 은 출력층의 유니트 수이다. 그리고 $\phi_j = s_j / \sum_{j=1}^h s_j$ 이다.

퍼지 방사형 함수망의 입출력 관계 u 는 다음과 같은 행렬형태로 표현될 수 있다.

$$u = \Phi(x)W \quad (3)$$

여기서, $W = [w_{11} w_{12} \dots w_{1h} \dots w_{i1} w_{i2} \dots w_{ih} \dots w_{m1}]$

$w_{m2} \dots w_{mh}]^T$ 는 $mh \times 1$ 가중치 벡터이고

$$\phi = \begin{bmatrix} \phi_1 & \phi_2 & \dots & \phi_h & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & \phi_1 & \phi_2 & \dots & \phi_h & \dots & 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \dots & \phi_1 & \phi_2 & \dots & \phi_h & \dots \end{bmatrix}$$
는 $m \times mh$ 행렬이다.

퍼지 시스템과 신경망은 각각 식(3)과 같은 형태를 갖는다. 즉, 지지함수 ϕ_j 는 퍼지 시스템에서 Wang^[11]이 서술한 퍼지 기준 함수(fuzzy basis function)와 신경망에서 시그모이드 함수나 방사형 함수와 동일한 식이다. 따라서 식(3)은 퍼지시스템과 신경망을 포함하는 지능 제어기의 일반화된 입력력 관계를 표현한다. 그러므로 전문가의 예비지식으로 부터 퍼지 규칙을 구성하였다면 이러한 퍼지 규칙을 지지함수 ϕ_j 로 쉽게 규합할 수 있다. 예로서, 2입력-1출력 형태의 퍼지 시스템을 고려할때 각 입력에 대한 퍼지 변수 A_1, A_2 를 다음과 같은 방사형 함수로 정의되었다고 가정한다.

$$\mu_{A_1}(x_1) = \exp \left[-\frac{(x_1 - \xi_1)^2}{\sigma_1^2} \right],$$

$$\mu_{A_2}(x_2) = \exp \left[-\frac{(x_2 - \xi_2)^2}{\sigma_2^2} \right]$$

이때 임의의 퍼지 규칙의 활성 정도 또는 소속정도는 다음과 같이 계산될 수 있다.

$$\begin{aligned} \mu_{A_1}(x_1) \mu_{A_2}(x_2) &= \exp \left[-\frac{(x_1 - \xi_1)^2}{\sigma_1^2} - \frac{(x_2 - \xi_2)^2}{\sigma_2^2} \right] \\ &= \exp \left[-\frac{a_1(x_1 - \xi_1)^2 + a_2(x_2 - \xi_2)^2}{\sigma^2} \right] \quad (4) \\ &= \exp \left[-\frac{(x - \xi)^T A (x - \xi)}{\sigma^2} \right] \end{aligned}$$

여기서, $\sigma^2 = a_1 \sigma_1^2 + a_2 \sigma_2^2$, $x = [x_1, x_2]^T$, $\xi = [\xi_1, \xi_2]^T$, 이고 $A = \text{diag}(a_1, a_2)$.
 상기의 식(4)는 식(2)와 동일하므로 지지함수 ϕ_j 는 전문가의 경험으로 부터 얻어진 퍼지 규칙으로 구성될 수 있다.

III. 로봇 매니플레이터의 적응 퍼지-신경망 제어시스템 설계

본 절에서는 II장에서 서술한 퍼지 신경망을 이용하여 적응 학습 제어시스템을 설계한다. 퍼지 신경망 제어를 위한 학습 규칙은 직접 적응 제어 방식(그림 2)에 의해 유도된다. 기존의 적응 제어 이론에서 직접 적응 제어 방식은 측정된 플랜트의 출력 y 와 플랜트의

기준 모델 출력 y_d 로 부터 얻어지는 오차를 최소화하도록 제어량 u 를 계산한다. 직접 적응 제어 방식은 명확한 플랜트의 모델에 의존하지 않고 온라인 플랜트 식별(identification)을 필요치 않는다. 그리고 기준 모델(reference model)은 기준량 r 이 주어졌을 때 실제 플랜트의 연속된 시간상에서의 목표 출력치를 제시함으로써 제어 목표를 구체화해주는 역할을 담당한다. 이때 기준 모델은 실제 플랜트의 특성(즉, rank와 같은 구조적인 특성)을 고려하여 설계된다.

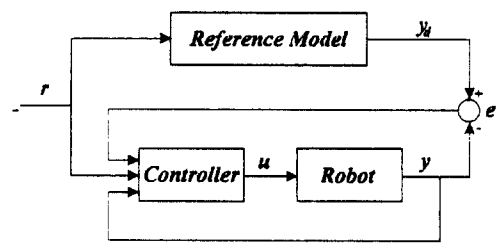


그림 2. 직접 적응 제어시스템
 Fig. 2. Direct adaptive control system.

1. 로봇 매니플레이터
 m 자유도를 갖는 로봇 매니플레이터는 다음과 같은 Lagrange식으로 표현된다.

$$M(q)\ddot{q} + c(q, \dot{q}) + g(q) = u \quad (5)$$

여기서, $q \in R^m$, $\dot{q} \in R^m$ 는 각각 로봇의 관절 각도와 각속도 벡터이다. 그리고 $u \in R^m$ 은 제어입력 벡터, $M(q) \in R^{m \times m}$ 은 관성행렬, $c(q, \dot{q}) \in R^m$ 는 원심력과 코리올리항을 포함하는 벡터, $g(q) \in R^m$ 는 중력 벡터이다.

액추에이터 동적특성을 무시하고 관절각도와 각속도는 측정 가능하다고 가정하고, $y = \text{col}(q, \dot{q}) \in R^n$, $n = 2m$ 로 정의하면 식(5)는 다음과 같이 표현된다.

$$\dot{y} = F(y) + G(y)u \quad (6)$$

여기서,

$$F(y) = \begin{bmatrix} y_{m+1} \\ y_{m+2} \\ \vdots \\ y_{2m} \\ \hline -M^{-1}(c+g) \end{bmatrix} \in R^n,$$

$$G(y) = \begin{bmatrix} 0 \\ \hline -M^{-1} \end{bmatrix} \in R^{n \times m}.$$

상기의 식에서 $F(y)$ 와 $G(y)$ 는 a, \dot{q} 의 복잡한 비선형 함수이다. 비선형 함수 $F(y)$ 와 $G(y)$ 는 불확실하다고 가정되어진다.

그리고 $G(y)$ 는 다음과 같은 속성을 갖는다.

$$G(y) = I_o M^{-1} \quad (7)$$

여기서, $I_o = [0 : I_m]^T$ 이고 M^{-1} 은 positive definite $m \times m$ 대칭행렬이다. 또한 M^{-1} 은 $\|M^{-1}\| < M_{\max}$ 으로 한정되어 있다고 가정된다.

2. 직접 적응 퍼지 신경망 제어시스템

그럼 2와 관련하여 직접 적응 퍼지 신경망 제어시스템은 다음과 같은 식의 집합으로 표현할 수 있다.

$$\dot{y}_d = A y_d + B r, \quad y_d \in R^n, \quad r \in R^m \quad (8)$$

$$\dot{y} = F(y) + G(y) u, \quad y \in R^n, \quad u \in R^m \quad (9)$$

식(8)과 (9)는 기준 모델과 비선형 로봇트 매니플레이터의 상태방정식을 나타낸다. 쌍 (A, B) 는 안정화될 수 있고 A 는 Hurwitz $n \times n$ 행렬이다. 기준 모델은 원하는 수행특성 즉 상승시간, 오버슈트의 특성을 설계자에 의해 최소의 매개변수들(자연주파수 ω 과 감쇠비 ζ)로 다음과 같이 정의될 수 있다.

$$A = \begin{bmatrix} 0 & I_m \\ -\Omega_0 & -\Omega_1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \Omega_0 \end{bmatrix}$$

여기서 $\Omega_0 = \text{diag}(\omega_i^2)$, $\Omega_1 = \text{diag}(2\zeta_i \omega_i)$, $\omega_i > 0$, $\zeta_i > 0$.

퍼지 신경망 제어기는 식(3)으로 부터 다음과 같이 주어진다.

$$u = \Phi(x) W \quad (10)$$

여기서, $x = \text{col}(r, y)$ 는 제어기의 입력 벡터이다.

기준 모델과 로봇트 매니플레이터의 출력 오차 $e = y_d - y$ 를 정의하면 식(8)과 (9)로 부터 다음과 같은 상태 오차 방정식을 얻을 수 있다.

$$\begin{aligned} \dot{e} &= A e + I_o M^{-1} (G^+ (A y + B r - F) - u) \\ &= A e + I_o M^{-1} (u_f - u) \end{aligned} \quad (11)$$

여기서, $u_f = G^+ (A y + B r - F)$ 이고, $G^+ = (G^T G)^{-1} G^T$ 이며 G^+ 는 G 의 Penrose Pseudo-inverse이다.

3. 적응 학습 규칙 유도

식(11)의 비선형항 u_f 는 비선형 지지함수(ϕ_f)를

예비지식으로 알고 있다면 지지함수들의 선형 결합에 의해서 다음과 같이 표현될 수 있다.

$$u_f = \sum_{j=1}^h w_{fj} \phi_{fj}, \quad \text{또는} \quad u_f = \Phi_f W_f \quad (12)$$

상기의 식(12)에서 지지 함수 ϕ_{fj} 는 Sanner가 제시한 샘플링 이론에 근거한 방식^[12]이나 퍼지 클러스터링 방식에 의해 구할 수 있다. 비선형 지지함수(ϕ_{fj})를 알고 있고 W_f 는 모른다고 가정하면 퍼지 신경망의 출력 u 를 다음과 같이 설정할 수 있다.

$$u = \Phi_f W \quad (13)$$

식(11)은 다음과 같이 다시 정리될 수 있다.

$$\begin{aligned} \dot{e} &= A e + I_o M^{-1} (\Phi_f W_f - \Phi_f W) \\ &= A e + I_o M^{-1} \Phi_f \tilde{W} \end{aligned} \quad (14)$$

여기서, $\tilde{W} (= W_f - W)$ 는 가중치 오차를 의미한다. 적응 학습 규칙을 유도하기 위해서 다음과 같은 Lyapunov 함수를 정의했을 때,

$$V = e^T \Lambda e + \tilde{W}^T \Gamma^{-1} \tilde{W} \quad (15)$$

Γ 는 positive definite 대각 행렬이고 Λ 는 식(16)의 Lyapunov 방정식을 만족하는 positive definite 대칭 행렬이다.

$$A^T \Lambda + \Lambda A = -Q \quad (16)$$

여기서, Q 는 positive definite 행렬이다.

V 의 도함수는 식(14)과 식(16)를 이용하여 다음과 같이 유도된다.

$$\begin{aligned} \dot{V} &= e^T A^T \Lambda e + e^T \Lambda A e \\ &\quad + (I_o M^{-1} \Phi_f \tilde{W})^T \Lambda e + e^T \Lambda (I_o M^{-1} \Phi_f \tilde{W}) + 2 \tilde{W}^T \Gamma^{-1} \dot{\tilde{W}} \\ &= e^T (A^T \Lambda + \Lambda A) e + 2 \tilde{W}^T \Phi_f^T M^{-T} I_o^T \Lambda e + 2 \tilde{W}^T \Gamma^{-1} \dot{\tilde{W}} \\ &= -e^T Q e + 2 \tilde{W}^T (\Phi_f^T M^{-T} I_o^T \Lambda e + \Gamma^{-1} \dot{\tilde{W}}) \end{aligned} \quad (17)$$

상기의 식에서 $\dot{\tilde{W}}$ 를 다음과 같이 선택하면

$$\dot{\tilde{W}} = -\Gamma \Phi_f^T M^{-T} I_o^T \Lambda e \quad (18)$$

식(17)는 다음과 같이 정리된다.

$$\dot{V} = -e^T Q e < 0 \quad (19)$$

상기의 식(15)와 (19)로 부터 e 와 W 는 한정되고 \dot{V} 는 연속함수이기 때문에 $\lim_{t \rightarrow \infty} \dot{V} = 0$ 을 의미한다. 그러므

로 $\lim_{t \rightarrow \infty} e = 0$ 이다.

적용 학습 규칙은 식(18)로 부터 구할 수 있다. 즉, $\dot{W} = \dot{W}_f - \dot{W} = -\dot{W}$ 이므로 \dot{W} 는 다음과 같이 유도된다.

$$\dot{W} = \Gamma \Phi_f^T M^{-T} I_0^T \Lambda e \quad (20)$$

4. 적용 학습 제어의 강건성

식(20)의 학습 규칙은 Φ_f 와 M^{-1} 을 알고 있다는 가정하에서 유도되었다. 비선형함수 u_f 는 많은 지지함수 ϕ_{f_i} 들로 표현되기 때문에 이러한 지지함수를 바탕으로 퍼지 신경망 제어를 구성하는 것은 실제 적용에 있어 용의하지 못하다. 따라서 퍼지 신경망 제어기는 제한된 지지함수 ϕ_{f_i} 들에 의해서 구성된다. 실제 구현 가능한 퍼지 신경망을 다음과 같이 u^* 로 정의한다.

$$u^* = \Phi W^* \quad (21)$$

여기서, Φ 는 제한된 지지함수 ϕ_{f_i} 들로 구성된 행렬이고 W^* 는 u_f 와 u^* 와의 최소 근사화 오차를 갖도록 하는 최적 매개변수이다.

그러므로 u_f 와 u^* 와의 최소 근사화 오차 e_f 가 식(22)와 같이 존재할 수 있다.

$$e_f = u_f - u^* \text{ 또는 } u_f = u^* + e_f \quad (22)$$

또한 실제 M^{-1} 은 근사추정치 \hat{M}^{-1} 에 불확실한 항 ΔM^{-1} 이 포함될 수 있으므로 식(23)과 같이 나타낼 수 있다.

$$M^{-1} = \hat{M}^{-1} + \Delta M^{-1} \quad (23)$$

식(22)과 식(23)로 부터 오차 상태방정식은 다음과 같이 다시 표현된다.

$$\begin{aligned} \dot{e} &= Ae + I_0(\hat{M}^{-1} + \Delta M^{-1})(u_f - u) \\ &= Ae + I_0(\hat{M}^{-1} + \Delta M^{-1})(u^* - u + e_f) \\ &= Ae + I_0(\hat{M}^{-1} + \Delta M^{-1})(\Phi W^* - \Phi W + e_f) \\ &= Ae + I_0 \hat{M}^{-1} \Phi \dot{W} + I_0(M^{-1} e_f + \Delta M^{-1} \Phi \dot{W}) \end{aligned} \quad (24)$$

여기서, $\dot{W} = W^* - W$.

Ω 를 다음과 같이 정의하면

$$\Omega = I_0(M^{-1} e_f + \Delta M^{-1} \Phi \dot{W}) \quad (25)$$

식(24)은 다음과 같이 나타낼 수 있다.

$$\dot{e} = Ae + I_0 \hat{M}^{-1} \Phi \dot{W} + \Omega \quad (26)$$

Lyapunov 함수를 식(27)과 같이 정의했을 때

$$V = e^T \Lambda e + \dot{W} \Gamma^{-1} \dot{W} \quad (27)$$

V의 도함수는 식(16)과 식(26)을 이용하여 다음과 같이 유도된다.

$$\begin{aligned} \dot{V} &= e^T A^T \Lambda e + e^T \Lambda A e + 2e^T \Lambda \Omega \\ &\quad + (I_0 \hat{M}^{-1} \Phi \dot{W})^T \Lambda e + e^T \Lambda (I_0 \hat{M}^{-1} \Phi \dot{W}) + 2\dot{W}^T \Gamma^{-1} \dot{W} \\ &= e^T (A^T \Lambda + \Lambda A) e + 2\dot{W}^T \Phi^T \hat{M}^{-T} I_0^T \Lambda e + 2\dot{W}^T \Gamma^{-1} \dot{W} + 2e^T \Lambda \Omega \\ &= -e^T Q e + 2\dot{W}^T (\Phi^T \hat{M}^{-T} I_0^T \Lambda e + \Gamma^{-1} \dot{W}) + 2e^T \Lambda \Omega \end{aligned} \quad (28)$$

상기의 식에서 W의 적용 학습 규칙을 다음과 같이 정의한다.

$$\dot{W} = -\Gamma \Phi^T \hat{M}^{-T} I_0^T \Lambda e \text{ 또는 } \dot{W} = \Gamma \Phi^T \hat{M}^{-T} I_0^T \Lambda e \quad (29)$$

그러므로 식(28)는 다음과 같이 정리된다.

$$\dot{V} = -e^T Q e + 2e^T \Lambda \Omega \quad (30)$$

식(30)에서 오차 e가 Ω 에 비해 상대적으로 클때 $\dot{V} < 0$ 이다. 그러므로 e는 한정되어 진다. 또한 $\dot{V} > 0$ 일 때 오차 e의 범위는 다음과 같이 주어진다.

$$\|e\| \leq 2 \frac{\lambda_{\max}(\Lambda)}{\lambda_{\min}(Q)} \|\Omega\| = e_{\max} \quad (31)$$

여기서, $\lambda_{\max}(\Lambda)$ 와 $\lambda_{\min}(Q)$ 는 각각 Λ 의 최대 고유치, Q의 최소 고유치이다.

식(31)은 불확실한 항이 포함될 경우에도 오차 e가 한정되어지는 것을 의미한다. 즉 $\lim_{t \rightarrow \infty} e = 0$ 를 만족할 수 없을 지라도 e는 최대 한계 e_{\max} 범위내에 존재하게 된다.^[13]

한계 범위 e_{\max} 는 크기는 $\|\Omega\|$ 에 비례한다. 식(25)로 부터 $\|\Omega\|$ 는 다음과 같이 한계범위를 갖는다.

$$\begin{aligned} \|\Omega\| &\leq \|I_0 \Delta M^{-1} \Phi \dot{W}\| + \|I_0 M^{-1} e_f\| \\ &\leq \|I_0 \Delta M^{-1} \Phi \dot{W}\| + \|I_0\| \|M^{-1}\| \|e_f\| \end{aligned} \quad (32)$$

상기의 식에서 첫번째항은 근사추정치 \hat{M}^{-1} 에 대한 불확실한 항 ΔM^{-1} 을 포함한다. 식(29)의 적용 규칙에 의해 $\dot{W} = W^* - W \approx 0$ 이 될때 거의 무시될 수 있다. 두번째항은 u_f 와 u^* 와의 최소 근사화 오차 e_f 를 포함한다. 이러한 오차는 퍼지 신경망의 근사화 능력에 의해 좌우된다. 퍼지 신경망 제어기를 충분한 지지함수

ϕ_j 들에 의해 구성한다면 근사화 오차 e_f 를 줄일 수 있고 최대 한계 e_{max} 를 제한할 수 있다.

IV. 모의 실험

본 절에서는 III장에서 얻어진 결과를 바탕으로 제안된 방식의 효율성을 검토하기 위해 2 자유도를 갖는 로봇트 매니플레이터의 적용예를 보인다. 2 자유도 로봇트 매니플레이터의 M, c, g의 특성은 다음과 같이 주어진다.

$$M(q) = \begin{bmatrix} l_1^2(m_1+m_2)+m_2l_2^2+2m_2l_1l_2\cos(q_2) & m_2l_2^2+m_2l_1l_2\cos(q_2) \\ m_2l_2^2+m_2l_1l_2\cos(q_2) & m_2l_2^2 \end{bmatrix},$$

$$c(q, \dot{q}) = \begin{bmatrix} -2m_2l_1l_2\sin(q_2)\dot{q}_1\dot{q}_2 - m_2l_1l_2\sin(q_2)\dot{q}_2^2 \\ m_2l_1l_2\sin(q_2)\dot{q}_2^2 \end{bmatrix},$$

$$g(q) = \begin{bmatrix} (m_1+m_2)gl_1\cos(q_1)+m_2gl_2\cos(q_1+q_2) \\ m_2gl_2\cos(q_1+q_2) \end{bmatrix},$$

여기서 m_1, m_2 , 그리고 l_1, l_2 는 각각 링크 1, 2의 질량 및 길이이다. 모의 실험을 위해 $m_1=0.7, m_2=0.5, l_1=0.5, l_2=0.5$ 로 정하였다.

상기의 식으로부터 상태변수 $y = [q_1 \ q_2 \ \dot{q}_1 \ \dot{q}_2]^T$ 로 정의하여 식(9)과 같은 형태로 표현하면 F, G는 다음과 같다.

$$F(y) = [y_3 \ y_4 \ f_3 \ f_4]^T,$$

$$G(y) = I_o M^{-1} = \begin{bmatrix} 0 \\ \dots \\ I \end{bmatrix} \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix}$$

여기서,

$$f_3 = \frac{[-m_2gl_1l_2^2(m_1+m_2)\cos(y_1)+m_2^2gl_1l_2^2\cos(y_1+y_2)\cos(y_2) + m_2^2l_1^2l_2^2\sin(y_2)\cos(y_2)y_3^2 + m_2^2l_1l_2^2\sin(y_2)(y_3+y_4)^2]}{d},$$

$$f_4 = \frac{[m_2gl_1l_2^2(m_1+m_2)\cos(y_1)-m_2^2gl_1l_2^2\cos(y_1+y_2)\cos(y_2) + \sin(y_2)(m_2(m_1+m_2)gl_1^2l_2^2\sin(y_1)-m_2^2l_1l_2^2(y_3+y_4)^2) - \sin(y_2)(m_2(m_1+m_2)l_1l_2y_3^2+m_2^2l_1^2l_2^2\cos(y_2)(y_3+y_4)^2)]}{d}$$

$$g_{11} = m_2l_2^2/d, \quad g_{12} = -(m_2l_2^2+m_2l_1l_2\cos(y_2))/d, \quad g_{21} = g_{12},$$

$$g_{22} = ((m_1+m_2)l_2^2+m_2l_2^2+2m_2l_1l_2\cos(y_2))/d,$$

$$d = m_2(m_1+m_2)l_1^2l_2^2 - m_2^2l_1^2l_2^2\cos^2(y_2).$$

그리고 식(8)에서 기준 모델의 A, B 행렬은 일반화된 상태오차의 감속속도들을 고려하여 $\omega_i=1, \zeta_i=1$ 로 두었다. 식(16)에서 Q를 4x4 단위행렬로 정하였다. 퍼지 신경망 제어기의 입력 x는 다음과 같다.

$$x = [r_1/N_1 \ r_2/N_2 \ y_1/N_3 \ y_2/N_4 \ y_3/N_5 \ y_4/N_6]^T.$$

여기서, N_k 는 각각의 입력을 $[-1, 1]$ 사이로 정규화하기 위한 환산계수(scaling factor)이며 최대 입력 범위를 고려하여 $N_1=N_2=1.5, N_3=N_4=1.0, N_5=N_6=2.0$ 으로 정하였다.

그리고 퍼지 신경망 제어기의 초기 구성은 II장에서 서술한 바와 같이 각 입력의 퍼지 방사형 소속함수를 2개씩 할당하여 지지함수의 갯수를 64개로 정하였다. 이때 지지함수 ϕ_j 의 중심 ξ_j 들은 -1 또는 1로 초기화하고 폭 $\sigma = 0.8$ 로 그리고 가중행렬 A의 요소들(a_{jk})은 모두 1로 두었다. 또한 중간층과 출력층과의 가중치 w_{ij} 는 $-0.00001 \sim 0.00001$ 사이의 값으로 랜덤하게 정하였다. 식(29)의 적응 학습 규칙에서 $\Gamma = \text{diag}(5, 5, \dots, 5)$ 로 두었고 실제 M^{-1} 의 근사추정치 M^{-1} 은 $M^{-1} = \begin{bmatrix} 5 & -10 \\ -10 & 30 \end{bmatrix}$ 으로 정하였다. 가중치 w_{ij} 는 식(29)에 의해 온라인으로 조정되어 진다.

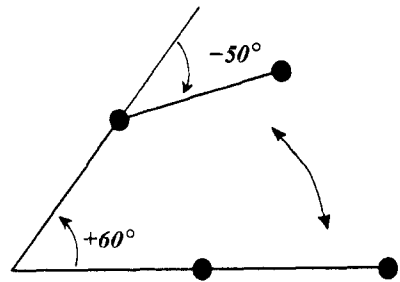
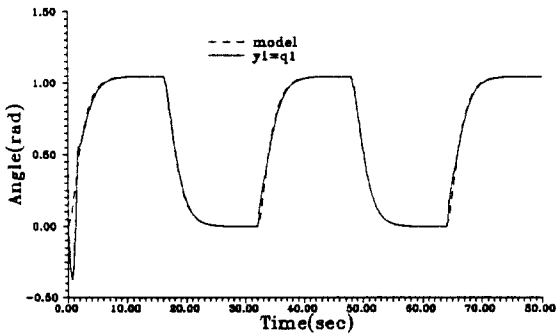


그림 3. 로봇트 매니플레이터의 초기 및 최종 링크 방위

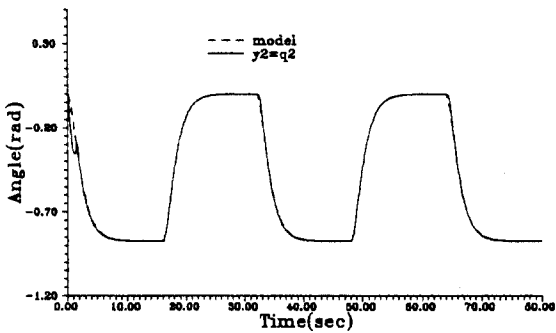
Fig. 3. Initial and final link orientation of robot.

제안된 방식의 제어성능을 고찰하기 위하여 기준 입력 r_1, r_2 는 그림 3과 같이 로봇트의 초기 및 최종 링크 방위를 반복하도록 주어진다. 그림 4와 5는 각각 퍼지 신경망 제어기에 의한 첫번째 학습시와 10번째 학습시 링크 1과 링크 2의 추종 결과와 추종 오차를 보인다. 그림 4에서 보이는 바와 같이 첫번째 적응 학습에서 초기 오차가 존재할지라도 퍼지 신경망 제어기에 의해 로봇트 매니플레이터가 기준 모델의 출력을 점근적으로 추종함을 볼 수 있다. 그림 5에서 10번째 적응 학습 시에는 초기 온라인 적응 학습 제어의 추종 결과보다 향상된 성능을 보이고 최대 오차 e_{max} 가 매우 적음을 알 수 있다. 수행된 모의 실험 결과로부터 제안된 적응 학습 제어 방식은 불확실한 로봇트 매니플레이터를

안정하게 제어할 수 있음을 볼 수 있다.



(a)



(b)

그림 4. 첫번째 학습시 추종 결과
(a) 링크 1 (b) 링크 2

Fig. 4. Tracking results during the first learning.
(a) link 1 (b) link 2

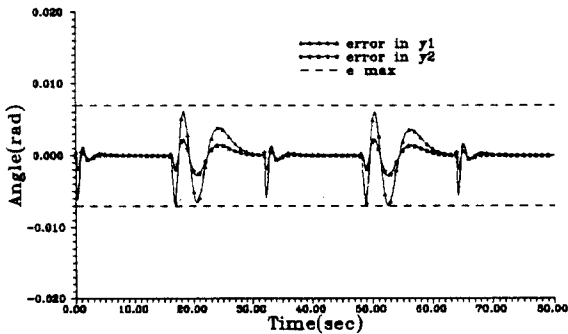


그림 5. 10번째 학습시 추종 오차

Fig. 5. Tracking errors during the 10th learning.

V. 결론

본 논문에서는 비선형 시스템인 로봇트 매니플레이

터를 제어하기 위해 적응 제어 방식과 지능제어 이론을 도입한 직접 적응 학습 제어 방식이 제안되었다. 그리고 퍼지 신경망을 이용하여 온라인 학습 제어시 안정성을 보장하도록 Lyapunov 안정성 이론을 바탕으로 적응 학습 규칙을 유도하였다. 적응 학습 규칙은 로보트 매니플레이터의 비선형 동적특성이 불확실하다는 가정하에서 적용되었다. 또한 제안된 퍼지 신경망 제어기의 강건성에 대해서 고찰하였다. 2 자유도 매니플레이터의 모의 실험으로부터 제안된 방식은 불확실한 로보트 매니플레이터를 효율적으로 온라인 학습 제어할 수 있음을 볼 수 있었다.

참고 문헌

- [1] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller-Part I, II," *IEEE Trans. Syst. Man. Cybern.*, vol. 20, no. 2, pp. 408-435, Mar./Apr. 1990.
- [2] A. Guez, J.L. Eilbert, and M. Kam, "Neural network architecture for control," *IEEE Control Syst. Mag.*, pp. 22-25, Apr. 1988.
- [3] J.S. Roger Jang, C.T. Sun, "Functional Equivalence Between Radial Basis Function Networks and Fuzzy Inference Systems," *IEEE Trans. on neural networks*, vol. 4, no. 1, Jan. 1993.
- [4] M.I. Jordan, "Generic constraints on underspecified target trajectories," *Proc. IEEE Int. Conf. on Neural Networks*, (Washington D.C.), pp. 217-225, 1989.
- [5] K.S. Narendra, K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. on neural networks*, vol. 1, no. 1, pp. 4-27, March 1990.
- [6] T. Hasegawa, S. Horikawa, T. Furuhashi, Y. Uchikawa, "On design of adaptive fuzzy controller using fuzzy neural networks and a description of its dynamical behavior," *Fuzzy Sets and Systems* 71, pp. 3-23, 1995.
- [7] T. Yabuta, T. Yamada, "Neural network controller characteristics with regard to adaptive control," *IEEE trans. on SMC*, vol. 22, no. 1, pp. 170-176, Jan./Feb. 1991.
- [8] A.G. Barto, R.S. Sutton, and C.W. An-

- derson, "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE Trans. on SMC*, vol. 13, pp. 834-846, 1983.
- [9] H. R. Berenji, P. Khedkar, "Learning and tuning fuzzy logic controllers through reinforcements," *IEEE Trans. on Neural Networks*, vol. 3, no. 5, pp. 724-740, 1992.
- [10] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are Universal Approximators," *Neural Networks*, vol. 2, pp. 359-366, 1989.
- [11] L.X. Wang, J.M. Mendal, "Fuzzy basis function, universal approximation, and orthogonal least squares learning," *IEEE Trans. on neural networks*, vol. 3, pp. 807-814, March 1992.
- [12] Robert M. Sanner, Jean-Jacques E. Slotine. "Gaussian networks for direct adaptive control," *IEEE Trans. on neural networks*, vol. 3, no. 6, Nov. 1992.
- [13] J.J. Craig, P. Hsu, and S.S. Sastry, "Adaptive control of mechanical manipulators," *Journal of Robotics Research*, vol. 6, no. 2, pp. 16-28, 1987.

 저 자 소 개

金 成 炫(正會員) 第 30卷 B編 第 4號 參照
 현재 중앙대학교 전자공학과 박사
 과정

崔 永 吉(正會員) 第 30卷 B編 第 4號 參照
 현재 중앙대학교 전자공학과 박사과정

金 容 虎(正會員) 第 31卷 B編 第 4號 參照
 현재 중앙대학교 전자공학과 박사
 과정

全 洪 兌(正會員) 第 31卷 B編 第 4號 參照
 현재 중앙대학교 전자공학과 교수