

DSP(TMS320C50)칩을 사용한 산업용 로봇의 적응-신경제어기의 실현

김용태*, 정동연*, 한성현**

Implementation of the Adaptive-Neuro Controller of Industrial Robot Using DSP(TMS320C50) Chip

Yong-Tae Kim*, Dong-Yean Jeong*, Sung-Hyun Han**

Abstract

In this paper, a new scheme of adaptive-neuro control system is presented to implement real-time control of robot manipulator using Digital Signal Processors. Digital signal processors, DSPs, are micro-processors that are particularly developed for fast numerical computations involving sums and products of variables. Digital version of most advanced control algorithms can be defined as sums and products of measured variables, thus it can be programmed and executed through DSPs. In addition, DSPs are as fast in computation as most 32-bit micro-processors and yet at a fraction of their prices. These features make DSPs a viable computational tool in digital implementation of sophisticated controllers. Unlike the well-established theory for the adaptive control of linear systems, there exists relatively little general theory for the adaptive control of nonlinear systems. Adaptive control technique is essential for providing a stable and robust performance for application of robot control. The proposed neuro control algorithm is one of learning a model based error back-propagation scheme using Lyapunov stability analysis method. The proposed adaptive-neuro control scheme is illustrated to be a efficient control scheme for the implementation of real-time control of robot system by the simulation and experiment.

Key Words : Adaptive-neuro control(적응-신경제어), Real-time(실시간), Digital Signal Processor(디지털 시그널 프로세서), Lyapunov stability(리아프노프 안정도)

1. 서 론

복잡한 동적특성을 갖는 시스템의 파라미터 변동, 또는

부하외란이 존재하는 동적환경에서 기존의 제어방법으로는 좋은 결과를 얻지 못하는 실정이다. 본 논문에서 채택한 적응 제어 기법은 컴퓨터 토크 제어 기법에서 부하

* 경남대학교 기계설계학과 대학원
** 경남대학교 기계자동화공학부
(shhan@kyungnam.ac.kr)

변동이나 알 수 없는 외란에도 변화하는 관성이나 마찰항을 추종하는 적응 제어 기법을 사용하고 있다. 신경회로망 기법은 시스템의 특성이 불확실 하더라도 신경망 학습기능으로 인하여 견실한 제어 성능을 얻고자 하는 것이 본 연구의 의도이다. 비선형 동적 시스템을 표현하고 제어할 수 있는 뉴럴 네트워크 구조를 사용할 때 요구되는 것들 중에 하나는 공학적인 문제에 직면하게 되는 동적 시스템을 얼마나 정확하게 모델링 하는가 하는 것이다. 그리고, 동적 시스템의 입·출력 변수를 어떻게 선정 할 것이냐 하는 문제가 따르게 된다. 동적 시스템 거동을 근사화 시키는 작업을 하기 위하여 뉴럴 네트워크 구조에서 피이드백 가중치가 필요하다는 것은 자명한 사실이다. 뉴럴 네트워크에 관한 연구 보고에서 그러한 네트워크는 오차 역전파라 불리는 세포 모델에 의한 학습방법을 제안하고 있다. 따라서, 뉴럴 네트워크의 입출력 관계는 가중치와 관련된 파라미터의 조합의 값으로 결정된다. 위에서 언급한 네트워크의 능력은 바로 동적 시스템의 거동을 표현할 수 있는 가중치의 조합이 존재하는냐에 달려있다고 해도 과언은 아닐 것이다. 그러면, 시간에 대하여 연속인 동적 시스템을 모델링 할 수 있는 뉴럴 네트워크를 구성하는 문제를 생각해 보자.

신경제어는 모든 계산이 각 뉴런에서 독립적으로 병렬 처리되므로 실시간 제어가 가능하고, 학습된 제어정보가 분산 저장되므로 입력정보에 잡음이 섞이는 경우에도 안정된 동작을 한다. 지금까지 발표된 많은 신경회로망 모델 중 제어 문제에 가장 많이 적용되는 모델은 다층 퍼셉트론이다.⁽¹⁾⁻⁽³⁾ 이러한 다층 신경회로망에서 원하는 기능을 수행하기 위한 적절한 가중치(weight)를 구하는 방법으로 오차 역전파(errorback-propagation) 학습⁽⁴⁾⁻⁽⁶⁾이 많이 사용되고 있다.

신경회로망 제어기법과 마찬가지로 적응제어기법에도 학습기능이 있다. 다시 말해서, 적응제어 시스템의 적응화와 신경회로망의 학습은 다소 흡사한 개념이다. 적응화는 한번 제어 동작에 바람직한 동작 즉, 시스템 다이내믹스를 고려하여 점근적인 안정(asymptotic stability)을 보장하는 것이고, 학습은 여러번 반복 시도하여 바람직한 제어동작을 할수 있도록 하는 방법이라 할 수 있다. 그리고, 학습에 관련된 수식도 비슷한 형태를 가지고 있다. 따라서, 이런 공통된 특성을 이용하여 오차를 줄이고자 하는 노력이 많이 시도되고 있다. 즉, 신경회로망 제어기에 적응 기법을 이용한 학습방법, 적응 제어기를 신경회로망으로 구성하는 법 등 많은 형태의 제어기가 개발되었다.

본 논문이 제안하는 제어기는 리아프노프 안정도 이론을 근거로 한 신경회로망 제어기가 전체 시스템의 중추로서 안정된 제어입력을 공급할 수 있도록 설계하고 적응제어기법으로 더욱더 보강된 형태의 제어기를 설계하는 것이 본 논문의 궁극적인 목적이며 자유도 스카라 로봇 매니퓰레이터의 위치 및 속도제어에 적용함으로써 그 효용성을 입증한다.

2. 적응-신경망 제어기

2.1 적응제어책

본 연구에서 적용된 적응제어의 기본구조는 컴퓨터-토크 제어알고리즘을 이용하여 파라미터 즉, 관성모멘트나 코리올리 원심력항을 추정해주는 갱신 법칙을 부가시킴으로써 파라미터 오차벡터가 제로가 되도록 하여 점근적인 안정을 추구하는 방법이다. 본 논문에서 접근한 적응제어는 질량이나 마찰계수와 같은 것을 추정하는 것이 아니기 때문에 정확한 제어 플랜트의 모델링을 요구하지 않는다는 것이 큰 특징이라 할 수 있다.

다음 식은 로봇 동역학과 서보 모터 및 기어비를 고려한 동적 방정식을 나타내고 있다.

$$R(\theta, \dot{\theta}, \ddot{\theta})\ddot{\theta} = H(\theta) \ddot{\theta} + V_m(\theta, \dot{\theta}) + G(\theta) + F(\dot{\theta}) \quad (1)$$

$$\tau = R(\theta, \dot{\theta}, \ddot{\theta})v \quad (2)$$

여기서 $R(\theta, \dot{\theta}, \ddot{\theta})$ 는 알려진 시간함수의 $n \times r$ 행렬, v 는 미지의 파라미터 벡터 그리고 H, V_m, G, F 는 각각 관성, 코리올리 및 원심력, 중력, 그리고 마찰력항을 나타낸다.

적응 제어 입력 토오크는 다음과 같이 표현될 수 있다.

$$\tau = \hat{H}(\theta)(\ddot{\theta}_d + K_v \dot{e} + K_p e) + \hat{V}_m(\theta, \dot{\theta}) \dot{\theta} + \hat{G}(\theta) + \hat{F}(\dot{\theta}) \quad (3)$$

추적오차의 정의로부터 식(3)은 다음과 같이 쓸 수 있다.

$$\tau = \hat{H}(\theta)(\ddot{e} + K_v \dot{e} + K_p e) + \hat{H}(\theta) \ddot{\theta} + \hat{V}_m(\theta, \dot{\theta}) \dot{\theta} + \hat{G}(\theta) + \hat{F}(\dot{\theta}) \quad (4)$$

식(1)과 식(4)로부터 다음과 같은 입력 토오크 식이 유도된다.

$$\tau = \hat{H}(\theta)(\ddot{e} + K_v \dot{e} + K_p e) + R(\theta, \dot{\theta}, \ddot{\theta})\hat{v} \quad (5)$$

여기서, \hat{v} 은 알 수 없는 미지 파라미터의 시변 추정을 나타내는 $n \times 1$ 벡터이다.

오차 방정식의 형태로 식을 정리하면 다음과 같이 된다.

$$\ddot{e} + K_v \dot{e} + K_p e = \hat{H}^{-1}R(\theta, \dot{\theta}, \ddot{\theta})\hat{v} \quad (6)$$

파라미터 오차는 다음과 같이 정의한다.

$$\tilde{v} = v - \hat{v} \quad (7)$$

오차 방정식을 상태 공간 형태로 다시 쓰면 다음과 같은 형태로 된다.

$$\dot{e} = Ae + BH^{-1}(\theta)R(\theta, \dot{\theta}, \ddot{\theta})\tilde{v} \quad (8)$$

추종 오차 벡터는 다음과 같이 정의한다.

$$e = \begin{bmatrix} e \\ \dot{e} \end{bmatrix} \quad (9)$$

$$B = \begin{bmatrix} O_n \\ I_n \end{bmatrix}, \quad A = \begin{bmatrix} O_n & I_n \\ -K_p & -K_v \end{bmatrix} \quad (10)$$

다음은 Lyapunov 안정성 해석을 통하여 추종 오차 벡터가 점근적인 안정을 하도록 하는 적응 제어칙을 선택하기 위하여 양의 한정인 리아프노브 함수를 다음과 같이 선정한다.

$$V = e^T P e + \tilde{v}^T \Gamma^{-1} \tilde{v} \quad (11)$$

여기서, P 는 $2n \times 2n$ 의 양의 한정 상수 대칭행렬이고, Γ 는 대각이며, 양의 한정인 $r \times r$ 행렬이다.

$$\Gamma = \text{diag}(y_1, y_2, \dots, y_r) \quad (12)$$

리아프노브 함수를 시간에 대하여 미분함으로써 다음과 같이 된다.

$$\begin{aligned} \dot{V} &= e^T \dot{P} e + \dot{e}^T P e + 2 \tilde{v}^T \Gamma^{-1} \dot{\tilde{v}} \\ [\dot{\tilde{v}}^T \Gamma^{-1} \tilde{v}] &= \tilde{v}^T \Gamma^{-1} \dot{\tilde{v}} \end{aligned} \quad (13)$$

식(8)을 식(13)에 대입하여 정리하면 다음과 같이 된다.

$$\begin{aligned} \dot{V} &= e^T P(Ae + B \hat{H}^{-1}(\theta)R(\cdot) \tilde{v}) + \\ & (Ae + B \hat{H}^{-1}(\theta)R(\cdot) \tilde{v})^T P e + 2 \tilde{v}^T \Gamma^{-1} \dot{\tilde{v}} \end{aligned} \quad (14)$$

스칼라 전치의 특성을 이용하여 항별로 조합하면 다음과 같은 식이 나온다.

$$\begin{aligned} \dot{V} &= -e^T Q e + \\ & 2 \tilde{v}^T (\Gamma^{-1} \dot{\tilde{v}} + R^T(\cdot) \hat{H}^{-1}(\theta) B^T P e) \end{aligned} \quad (15)$$

여기서, Q 는 리아프노브 함수를 만족하는 양의 한정 행렬이다.

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

\dot{V} 는 항상 최소한 음의 반한정이어야 안정하다는 사실에서 적응 갱신 법칙을 다음과 같이 유도 할 수 있다.

$$\dot{\tilde{v}} = -\Gamma R^T(\cdot) \hat{H}^{-1}(\theta) B^T P e \quad (17)$$

$$\dot{V} = -e^T Q e \quad (18)$$

\dot{v} 가 0이기 때문에 파라미터 추종 벡터 $\dot{\hat{\phi}}$ 를 적응 갱신 법칙에 대입하면 다음과 같은 최종 적응제어칙을 형성한다.

$$\dot{\tilde{v}} = -\Gamma R^T(\cdot) \hat{H}^{-1}(\theta) B^T P e \quad (19)$$

\dot{V} 는 ($A > 0$) 항상 음이면 오차와 점근적으로 영으로 간다. 위의 사실과 V 가 항상 시간에 따른 연속함수라면, 다음의 결론을 얻을 수 있다.

$$\lim_{t \rightarrow \infty} \dot{V} = 0 \quad (20)$$

$$\lim_{t \rightarrow \infty} e = 0 \quad (21)$$

각 관절의 가속도 $\ddot{\theta}_v$ 는 다음과 같이 실제적으로 수치 미분의 형태로 아래와 같이 계산될 수 있다.

$$\ddot{\theta}_d^{t+1} = \frac{\dot{\theta}_d^{t+1} - \dot{\theta}_d^t}{\Delta t} \equiv \frac{\dot{\theta}_d^t - \dot{\theta}_d^{t-1}}{\Delta t}$$

Fig. 1은 제안된 적응제어기 구조의 블록 선도를 나타내고 있다.

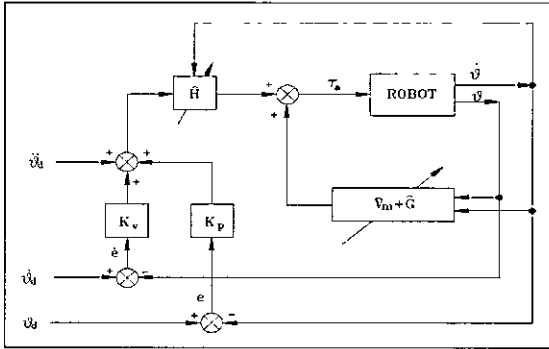


Fig. 1 Block diagram of the adaptive controller

2.2 신경회로망 설계

2.2.1 신경회로망의 구조

Fig. 2는 신경회로망 레귤레이터(neural network regulator)의 블록도이다. 각각의 링크에 다입력 단일출력(MISO)의 신경회로망으로 구성한다. 본 논문에서 사용한 신경회로망은 전체 시스템에 크게 영향을 미치는 제인을 보상해주는 가변 이득과 같은 역할을 한다. 신경회로망은 전체 시스템의 오차에 대해 빠른 적응성을 갖기 위하여 간단하게 설계한다. 신경회로망의 입력은 실제모델과 원하는 모델과의 오차이며 출력은 플랜트의 제어입력이다. 본 논문의 학습규칙은 입력층과 출력층 사이의 갱신 가중치 법칙을 Lyapunov 이론에 근거하여 정함으로써 학습된다.

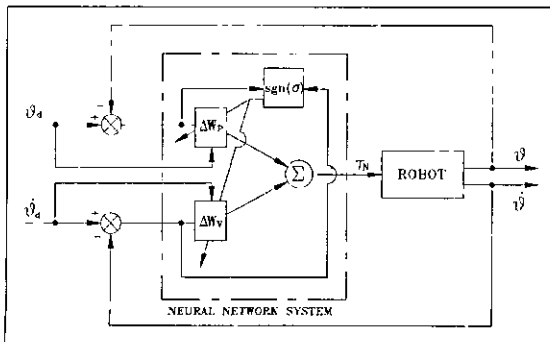


Fig. 2 The block diagram of neural network

$$\tau_n = x_1 w_1 + x_2 w_2 + \dots, x_n w_n = \sum_{i=1}^n X_i W_i \quad (22)$$

$$\tau_n(t) = \sum_{i=1}^n W_i X_i(t) \quad (23)$$

여기서 $x_i = [x_1(t), \dots, x_n(t)]$ 이며, $i = 1, \dots, n$ 는 i 번째 요소의 제어입력이다. W_i 는 가중치 요소이다.

2.2.2 신경망 제어책

기준모델 제어방법은 크게 3가지가 있는데 기울기 법, 리아프노브 안정성 이론 그리고, 수동 이론이 있다. 본 논문에서는 리아프노브 안정성 이론을 근거로 해석한다. 선형 시불변 시스템인 기준모델과 실제 플랜트의 동적모델로 되어있고, 플랜트의 출력이 기준모델의 출력을 추정하도록 제어하는 기법으로, 파라미터를 모르거나 환경의 변화에 적당한 제어기이다. 전방향(feedforward) 제어입력은 신경회로망의 원하는 입력과 시스템 출력의 차이이고, 부채환(feedback) 입력은 신경망 메카니즘(neural-network mechanism)에서 만들어진다. 제어대상은 4자유도 스카라 로봇으로 정하고, 각 링크의 기준 모델을 상태방정식으로 나타내면 다음과 같이 정할 수 있다.

$$\dot{X}_m = A_m X_m + B_m U_i \quad (24)$$

$$U_i = [u_i, \dots, u_n]^T \quad (25)$$

여기서,

$$A_m = \begin{bmatrix} 0, & I \\ -w_n^2, & -2\zeta w_n \end{bmatrix} B_m = \begin{bmatrix} 0 \\ w_n^2 \end{bmatrix}$$

그리고, 비선형 시변(NLTV)인 매니플레이터의 방정식은 일반적으로 다음과 같이 된다.

$$H(\theta)\ddot{\theta} + V_m(\theta, \dot{\theta}) + G(\theta) = \tau \quad (26)$$

이 식을 상태방정식으로 다시 정리하면 다음과 같다.

$$\dot{X}_p = A_p X_p + B_p U_p \quad (27)$$

$$A_p = \begin{bmatrix} 0, & I \\ -H^{-1} V_m, & -H^{-1} G \end{bmatrix} B_p = \begin{bmatrix} 0 \\ H^{-1} G \end{bmatrix}$$

Fig. 2로부터 제어출력이 다음과 같이 됨을 알 수 있다.

$$\tau_N = W_i X_i \quad (28)$$

τ_N 은 신경회로망 레귤레이터로부터 나오는 제어출력이다. 제어규칙을 유도하기 위해서는 Lyapunov 이론에 의해 이끌어 낸다. 식 (24), (27), (28)으로부터

$$\dot{X}_p(t) = A_p x_p(t) + B_p [W_i X_i] \quad (29)$$

$$\dot{X}_m(t) = A_m x_m(t) + B_m [W_i X_i] \quad (30)$$

여기서, W_i 는 출력층 가중치 행렬 이며, W_i^* 는 상수이다.

식(29)와 식(30)에서 다음과 같은 오차방정식이 유도된다.

[단, $A_p \approx A_m$]

$$e_a(t) \triangleq X_p(t) - X_m(t), \quad \varphi(t) \triangleq W_i(t) - \hat{W}_i(t) \\ \dot{e}_a(t) = A_m e_a(t) + B_p \varphi(t) X_i(t) \quad (31)$$

여기서, e_a 는 기준모델과 실제모델의 상태 오차이고, φ 는 평형점에서의 가중치 오차이다.

안정성 해석 및 뉴럴 제어규칙을 유도하기 위한 Lyapunov 함수 즉, 성능 함수를 최소화하는데 있어서 학습 알고리즘이 유도된다.

$$V(e_a, \varphi) \triangleq \frac{1}{2} [e_a^2 + |B_p| (\varphi^2)] \quad (32)$$

안정조건(stability condition)은 $\dot{V}(e, \varphi) < 0$ 이다.

$$\begin{aligned} \dot{V}(e_a, \varphi, \phi) &= e_a \dot{e}_a + |B_p| [\varphi \dot{\varphi}] \\ &= e_a [A_m e_a(t) + B_p \varphi(t) X_i(t)] + |B_p| [\varphi \dot{\varphi}] \\ &= A_m e_a^2 + B_p \varphi e_a X_p - |B_p| [\text{sgn}(B_p) \varphi e_a X_i] \\ &= A_m e_a^2 \leq 0 \end{aligned} \quad (33)$$

안정성(stability) 조건을 만족하기 위한 뉴럴 네트워크 제어규칙을 유도한다.

$$\dot{W}_i(t) = \dot{\varphi}(t) = -\eta \text{sgn}(B_p) e_a(t) X_i(t) \quad (34)$$

여기서, η 는 학습율을 나타낸다.

2.3 적응-신경 제어기 구성

전체 제어 알고리즘은 계산 토오크 방법(computed-torque method)의 기본 구조에 의한 적응 제어 기법(mechanism)으로 각 링크의 관성 항과 코리올리 원심력 항인 불확실한 파라미터(uncertain parameter)를 추정하여 실제 값에 근접한 파라미터 값을 산출하고, 신경망 제어로부터 기준 모델에서 제시된 상태파라미터와 실제 플랜트의 차를 학습 가중치 갱신(weight update) 법칙에 보내어 가중치가 갱신된다. 각각의 갱신 가중치가 궤적 계획(trjectory planning)에서 생성된 원하는 궤적(desired trajectory)과 실제 제어 대상인 플랜트(plant)의 궤적과의 오차 벡터와 대수적으로 곱해져서 신경망 제어 토오크를 산출하고 적응제어책에서 산출한 파라미터값이 최종제어책에 대입되어 적응-신경 제어 토오크를 산출하는 구조로 설계하였다. 제안된 제어기는 컴퓨터 토오크 방법의 장점을 충분히 살리고 단점을 보완하려는 의도에서 설계하였다.

$$\begin{aligned} \tau_c(t) &= \hat{H}'(t) [\ddot{\theta}(t) + W_{p1}(t)e(t) + W_{vi}(t)\dot{e}(t)] \\ &\quad + \hat{V}'(t) + \hat{C}'(t) \end{aligned} \quad (35)$$

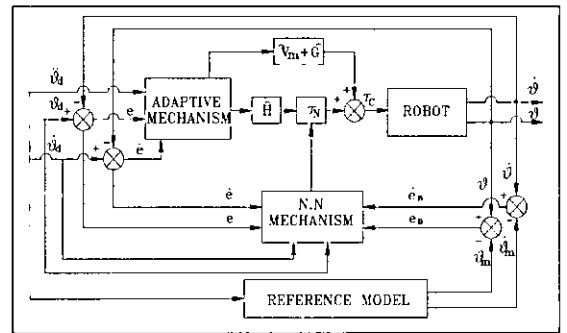


Fig. 3 The block diagram of adaptive-neuro controller

$$W_p(t) = \text{diag}[W_{p1}(t), W_{p2}(t), \dots, W_{pn}(t)]$$

$$W_v(t) = \text{diag}[W_{v1}(t), W_{v2}(t), \dots, W_{vn}(t)]$$

여기서, τ_c 는 적응-신경 제어기의 최종 제어 토오크를 나타내며 적응 제어책에 의한 파라미터 갱신 알고리즘은 식 (36)와 식 (37)에 나타나어 있다.

$$\hat{H}'(t+1) = \hat{H}'(t) + \lambda_i \int_0^t R_{ij}(t) / \hat{H}'(t)$$

$$[P_2 e_i(t) + P_3 \dot{e}_i(t)] dt \quad (36)$$

$$\begin{aligned} \hat{V}'(t+1) &= \hat{V}'(t) + \lambda_i \int_0^t R_{ij}(t) / \hat{H}'(t) \\ & [P_2 e_i(t) + P_3 \dot{e}_i(t)] dt \end{aligned} \quad (37)$$

여기서, $\hat{H}'(t+1)$ 과 $\hat{V}'(t+1)$ 은 각각 관성 항과 등가 점성 마찰을 추정하는 갱신 알고리즘이며, λ_i 은 i 번째 관절의 갱신비이고, P_2 와 P_3 은 다음 식에서 구할 수 있다.

$$P = \begin{bmatrix} P_1 I_n & P_2 I_n \\ P_3 I_n & P_4 I_n \end{bmatrix} = \frac{1}{2} \begin{bmatrix} (k_p + 1/2k_v) I_n & 1/2 I_n \\ 1/2 I_n & I_n \end{bmatrix}$$

여기서, P 는 대칭 양의 행렬이고, I_n 은 2×2 단위 행렬(identity matrix)이다.

신경망 제어칙(neural network control law)을 유도하기 위한 기준모델의 토오크 방정식은 다음과 같이 표현될 수 있다. 식(38)은 독립 관절 형태로 구한 로봇 동역학과 모터와의 사이에 기어비로써 결합된 선형화 부분을 나타낸 것이다.

$$U_{mi} = J^{eq_i} \ddot{\theta}_{di}(t) + B^{eq_i} \dot{\theta}_{di}(t) + F^{eq_i} \quad (38)$$

여기서, J^{eq_i} 는 관성모멘트의 선형요소와 모터의 부하 축으로의 등가 관성모멘트의 합을 나타내고, B^{eq_i} 는 모터 회전자에의 감쇠계수항, 역기전력에 의한 매니플레이터의 점성계수의 합을 나타내고, F^{eq_i} 는 로봇 매니플레이터와 모터의 마찰항을 합친 것을 나타낸다. 기준 모델의 토오크로 인한 파라미터 출력으로 가중치를 갱신하는 것은 가중치의 크기(scale)를 조정하게 되는 효과가 있다. 신경회로망의 가중치 증감을 결정해주는 알고리즘을 유도하기 위하여 식(6)의 오차 방정식 형태로 다시 쓰자.

$$\ddot{e} + K_v \dot{e} + K_p e = \hat{H}^{-1} R(\theta, \dot{\theta}, \ddot{\theta}) \tilde{v} \quad (39)$$

위의 식을 상태 방정식으로 나타내면 다음과 같이 정의된다.

$$\begin{aligned} \dot{X}^{e_{1i}} &= X^{e_{2i}} \\ \dot{X}^{e_{2i}} &= -K_{vi} X^{e_{2i}} - K_{pi} X^{e_{1i}} \\ & + \hat{H}^{-1} R(\theta, \dot{\theta}, \ddot{\theta}) \tilde{v} \end{aligned} \quad (40)$$

여기서, X^{e_i} 는 i 번째 관절의 오차 상태 벡터이다.

오차 상태 공간상에서 절환선(switching line)인 점근적인 안정 다항식 $\kappa_i(t)$ 을 선택하여 신경망 가중치 진행 방향을 제어 하게 될 조건을 다음과 같이 선정한다.

$$\begin{aligned} \kappa_i(t) &= C^T X^e(t) \\ &= C_{1i} X^{e_{1i}}(t) + C_{2i} X^{e_{2i}}(t) \leq 0 \end{aligned} \quad (41)$$

$$\sigma_i(t) = X^{e_{1i}}(t)(C_{1i} X^{e_{1i}}(t) + C_{2i} X^{e_{2i}}(t)) \quad (42)$$

여기서 σ 에 의하여 신경망 제어칙의 증감 방향을 결정하게 되며, C_{1i} 와 C_{2i} 는 절환 선의 기울기를 결정하는 상수이며 설계자가 적절하게 선정한다. C_{1i} 와 C_{2i} 의 값은 오차 상태 공간 상에서의 오차에 대한 절환 선의 기울기를 결정하는 값이므로 설계자가 신중하게 결정하여야 한다.

$$W_{pi}(t+1) = W_{pi}(t) + \eta_{pi} \int_0^t \text{sgn}(\sigma) |e_{ai} \theta_{di}| dt \quad (43)$$

$$W_{vi}(t+1) = W_{vi}(t) + \eta_{vi} \int_0^t \text{sgn}(\sigma) |\dot{e}_{ai} \dot{\theta}_{di}| dt \quad (44)$$

[단 $\eta_{pi} > 0, \eta_{vi} > 0$]

$$\text{sgn}(\sigma) = \begin{cases} 1 & \sigma > 0 \\ -1 & \sigma < 0 \end{cases} \quad (45)$$

여기서, 각각 $W_{pi}(t)$, $W_{vi}(t)$ 는 비례 계인과 미분 계인을 추종하는 가변 계인이다.

3. 관절 제어기 설계(H/W)

실시간 제어를 위하여 고속 수치 전용 chip인 TMS320C5X를 사용한 관절 제어기를 구성하였다.

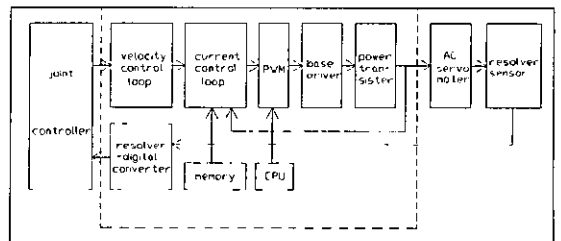


Fig. 4 AC joint driver

- 관절제어기로부터 관절구동부로 전달되는 신호 : 속도명령, 전류명령, 서보 ON/OFF명령, 서보 RESET 명령.
- 관절구동부로부터 관절제어기로 전달되는 신호 : 위치신호, 속도신호, error 신호.
- 관절제어기로부터 속도명령 계산 : 리졸버-디지털(R/D) 변환기에서 계산하여 12Bits 위치값과 아날로그 속도값으로 바꾸어 상위의 관절제어기로 공급한다.

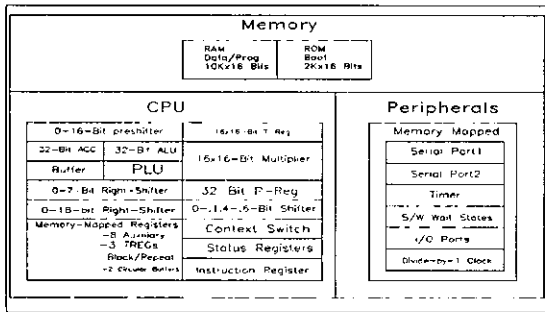


Fig. 5 The block diagram of TMS320C5X

4. 시뮬레이션 및 실험

4.1 시뮬레이션 및 결과

제안된 적응-신경제어 성능의 신뢰성을 확인하기 위하여 시뮬레이션을 수행하였다. 시뮬레이션은 실제 실험대상 로봇인 SCARA(SM5)를 대상으로 수행하였으며, 그 구조는 Fig. 6과 같으며, 시뮬레이션에 사용되어진 파라미터 및 조건은 Table 2 ~ Table 3에서 나타내고 있다. 시물

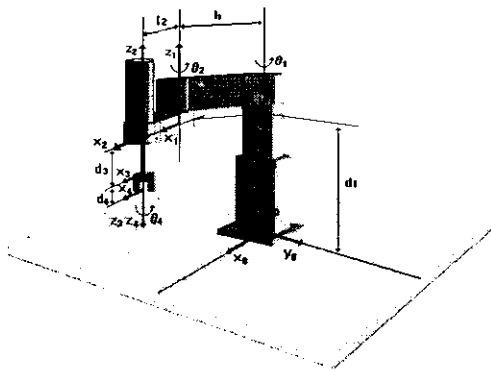


Fig. 6 Coordinates frame of a SCARA robot.

Table 2 The specification of SCARA robot

Link	Mass of Link(kg)	Length of Link(m)	Gear ration of joint
m1	15.067	l1	n1
m2	8.994	l2	n2
m3	3.0	d3	n3
m4	1.0	d4	n4

Table 3 The specification of motor

Motor	Torque coefficient (K.m/A)	Back Emf. coefficient (V/(rad/s))	Equivalent Resistance		
Ka1	21.48×10^{-2}	Kb1	214.85×10^{-3}	Ra1	1.5
Ka2	20.01×10^{-2}	Kb2	200.53×10^{-3}	Ra2	4.2
Ka3	20.01×10^{-2}	Kb3	200.53×10^{-3}	Ra3	9
Ka4	17.65×10^{-2}	Kb4	176.66×10^{-3}	Ra4	20

Table 4 The condition of simulation and emulation experiment

Initial Weight of 1, 2 joint	$w_{p1} = 0.0$	$w_{p2} = 0.0$	$w_{v1} = 0.0$	$w_{v2} = 0.0$
Initial Weight of 3, 4 joint	$w_{p3} = 0.0$	$w_{p4} = 0.0$	$w_{v3} = 0.0$	$w_{v4} = 0.0$
Renewals Cost of 1, 2 joint	$\lambda_1 = 0.001$		$\lambda_2 = 0.001$	
Renewals Cost of 3, 4 joint	$\lambda_3 = 0.001$		$\lambda_4 = 0.001$	
Learning rate of 1, 2 joint	$\eta_{p1} = 1.0$	$\eta_{p2} = 1.0$	$\eta_{v1} = 1.0$	$\eta_{v2} = 1.0$
Learning rate of 3, 4 joint	$\eta_{p3} = 1.0$	$\eta_{p4} = 1.0$	$\eta_{v3} = 1.0$	$\eta_{v4} = 1.0$
Initial Parameter estimate	$\hat{H}_1 = 2.5$	$\hat{H}_2 = 1.5$	$\hat{V}_{m1} = 0.5$	$\hat{V}_{m2} = 0.5$
Initial Parameter estimate	$\hat{H}_3 = 0.1$	$\hat{H}_4 = 0.02$	$\hat{V}_{m3} = 0.002$	$\hat{V}_{m4} = 0.001$
Switching line constant of 1, 2 joint	$C_{11} = 0.5$	$C_{21} = 0.2$	$C_{12} = 0.5$	$C_{22} = 0.2$
Switching line constant of 3, 4 joint	$C_{13} = 0.5$	$C_{23} = 0.2$	$C_{14} = 0.5$	$C_{24} = 0.2$
Sampling Time	1(ms)			
Integral Interval	0.01			

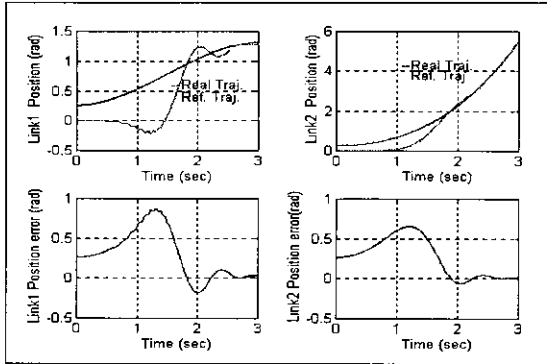


Fig. 7 Control performance of neuro controller for the position trajectory of each link after the first iteration

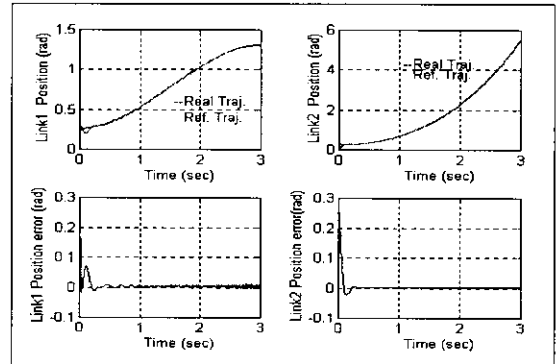


Fig. 9 Control performance of neuro controller for the position trajectory of each link after the sixth iteration

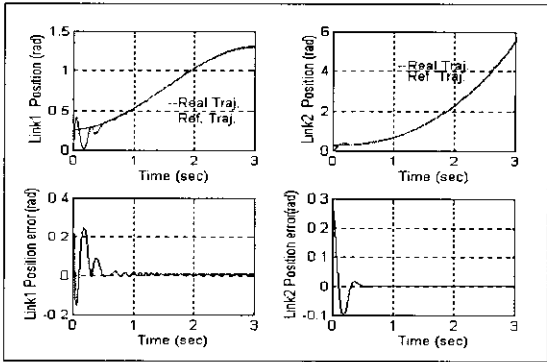


Fig. 8 Control performance of neuro controller for the position trajectory of each link after the fourth iteration

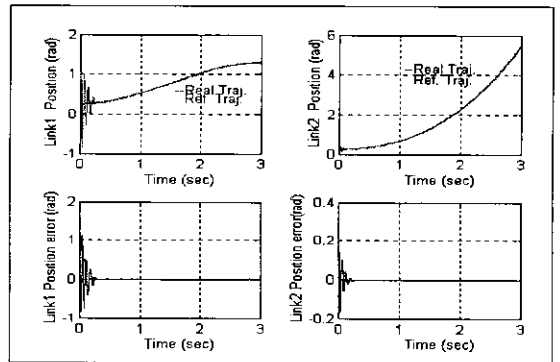


Fig. 10 Control performance of adaptive-neuro controller for the position trajectory of each link with parameter uncertainties 20(%)

레이션은 임의의 동일한 궤적에 대하여 신경제어기의 학습을 증가 시켜 가며 위치 추종 성능을 분석하였으며, 또한 제안된 적응-신경제어기에 파라미터 불확실성이 존재할 경우 제어 성능에 대하여 분석 하였다.

Fig. 7은 신경 제어기에서 1회 학습 결과 Link 1과 Link 2의 위치 추종 성능을 나타내고 있으며, Fig. 8은 4회 학습을 반복한 결과이며, Fig. 9는 6회 반복한 결과를 나타내고 있다. 그림에서와 같이 신경 제어기에서 반복회수가 증가함에 따라 Link 1과 Link 2의 위치 추종 성능이 향상됨을 확인할 수 있으며, 그 결과 또한 양호함을 확인할 수 있다. 그리고 Fig. 10은 파라미터 불확실성(20%)이 존재할 경우 제안된 적응-신경 제어기의 제어 성능을 나타내고 있다. 따라서, 시뮬레이션 결과 제안된 적응-신경 제어기의 타당성을 확인할 수 있다.

4.2 실험 및 결과

제안된 제어알고리즘을 실시간에 의해 실험적으로 검토하기 위해 스카라 로봇에 제어기로서 DSPs(digital signal processor)를 이용하여 제어기를 구성하였고, 이를 이용하여 2축에 대한 실시간 적응-신경제어를 수행하여 그 성능을 평가하였다. Fig. 11은 본 연구의 실험에서 사용된 시스템의 전체 구성도를 블록선도로 나타낸 것이다. 또한 Fig.12는 실제 실험장치를 나타내고 있으며, 실험을 수행한 결과는 Fig.13 ~ Fig.19에 나타내고 있다. Fig. 13와 Fig. 16은 기존에 내장되어 있는 PID 제어기법에 의한 Link 1과 Link 2의 위치 및 속도 추종 성능을 나타내고 있으며, Fig. 14와 Fig. 17은 제안된 적응-신경제어기법에 의한 Link 1과 Link 2의 위치 및 속도 추종 성능을 나타내고 있다. 그림에서와 같이 제안된 적응-신경제어기법을 이용한 제어 성능이 기존의 PID 제어기법보다

위치 추종 성능면에서 좋은 성능을 나타내고 있으며, 속도 추종 성능에서는 채털링이 급격히 감소됨을 확인할 수 있다.

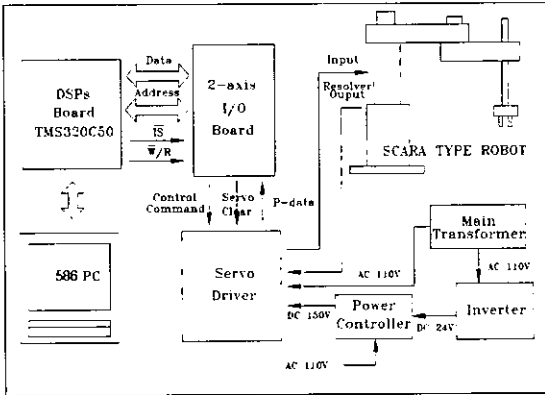


Fig. 11 The total control system of a SCARA robot

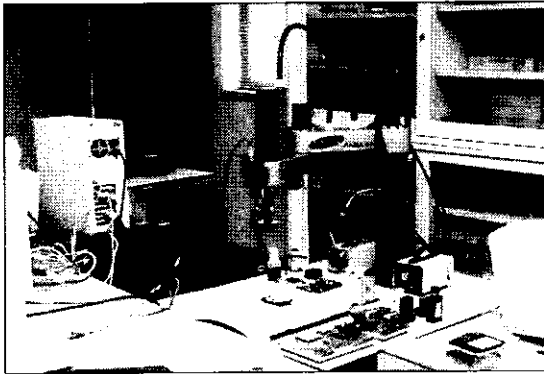


Fig. 12 Experiment set-up equipment

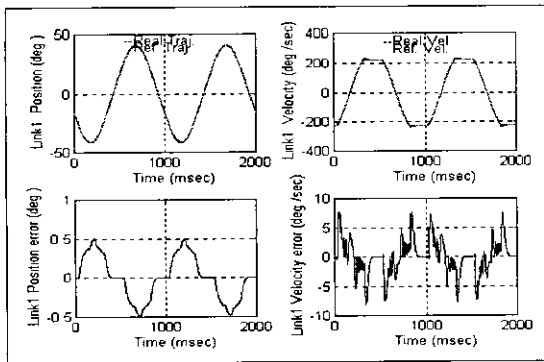


Fig. 13 Control performance of PID controller for the position and velocity trajectory of link 1 with no-load

Fig. 15와 Fig. 18은 제한된 적응-신경 제어 알고리즘에 파라미터 불확실성(20%)이 존재할 경우 Link 1과 Link 2의 위치 및 속도 추종성능을 나타내고 있으며, 그

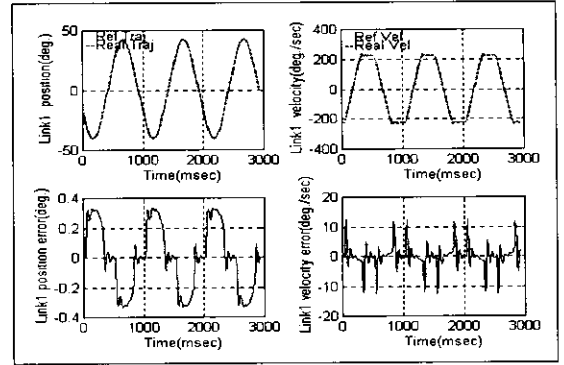


Fig. 14 Control performance of adaptive-neuro controller for the position and velocity trajectory of link 1 with no-load

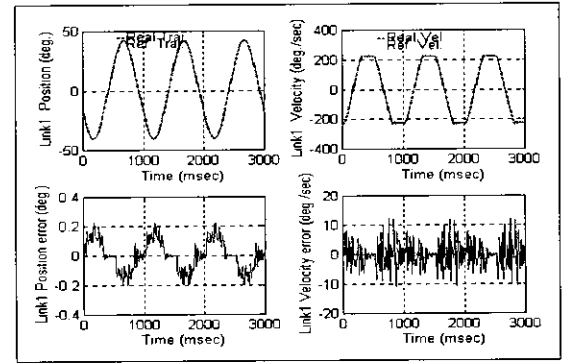


Fig. 15 Control performance of adaptive-neuro controller for the position and velocity trajectory of link 1 with inertia parameter uncertainties (20%)

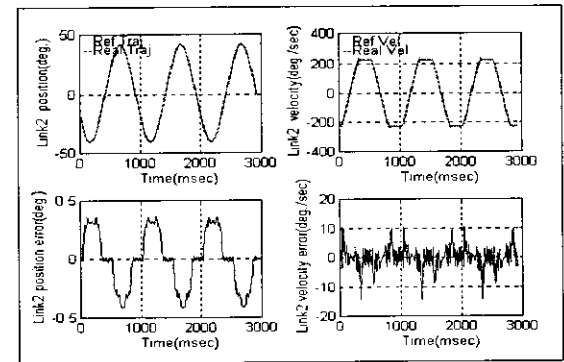


Fig. 16 Control performance of PID controller for the position and velocity trajectory of link 2 with no-load

결과 또한 양호함을 확인할 수 있다. Fig. 19는 파라미터 불확실성(10%)이 존재할 경우 Link 1과 Link 2의 토크 응답 특성을 나타내고 있다.

5. 결론

본 연구에서는 오차역전파 형태의 신경망을 주축으로 하여 비선형 시스템에 대한 DSPs를 이용한 새로운 구조의 적응-신경제어기를 설계하였다. 그리고 기존의 적응제어 시스템에 신경망의 장점을 충분히 살리려고 시도하였고, 두가지 제어방식의 장점을 이용하여 견실하고 실시간 제어에 적합한 관절 제어기를 개발하였다. 또한, 디지털 신호처리기인 TMS320C50의 어셈블러 제어 알고리즘(S/W)을 개발하고 TMS320C5x emulator를 스카라 로봇에 연결하여 시행한 실험을 통하여 제어된 적응-신경제어기의 견실성을 입증하였다. 실험결과 로봇이 외란을 받거나, 부하변동 및 기준궤적의 변동시에도 신속 정확한 제어 동작이 요구되는 곳에서 실시간으로 안정된 동작이 유지되었다. 따라서, 제안된 적응-신경제어기는 비선형 시스템의 온 라인 제어에의 효율적 적용 가능성을 제시하면서 향후 연구계획은 실험을 통해서 얻은 제어성능을 토대로 다양한 로봇 제어기에 적용가능성 및 상품화를 위한 신뢰성을 검증하고자 한다.

참고 문헌

- (1) Irfan. Ahmed,(Ed.). Digital Control Applications with the TMS320 Family. Selected Application Notes, Texas Instruments Inc., pp. 204-209, 1991.
- (2) S. Dubowsky and D.T. DesForges, "The Application of Model Reference Adaptation Control to Robot Manipulators," ASME J. Dyn. Syst., Meas., Contr., Vol. 101, pp. 193-200, 1979.
- (3) P.C.V. Parks, "Lyapunov Redesign of Model Reference adaptive Control System," IEEE Trans. Auto. Contr., Vol. AC-11, No.3, pp. 362-267, July 1966.
- (4) D. Psalitis, A. Sidiris, and A. Yamamura, "A Multi-layered Neural Network controller," IEEE Control Systems Magazine, pp. 17-21, April 1988.
- (5) P. Tomei, "Adaptive PD Controller for Robot Manipulators," IEEE Trans. Robotics and Automation, Vol. 7, No. 4, pp. 294-298, Aug 1991.
- (6) J. J. E. Slotine and W. Li, "Adaptive Manipulator Control - A case Study," Proc. IEEE Conf. Robotics and Automation, pp. 1392-1400, 1987.

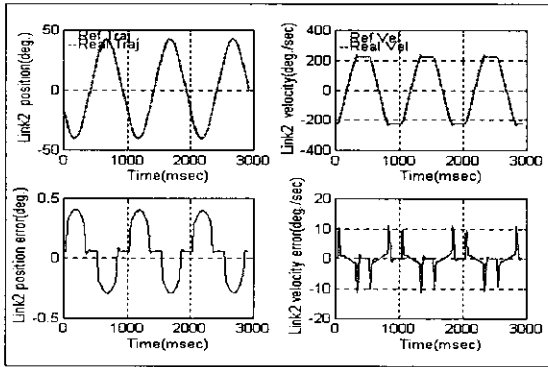


Fig. 17 Control performance of adaptive-neuro controller for the position and velocity trajectory of link 2 with no-load

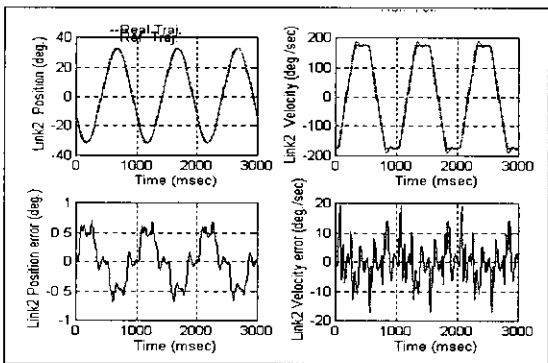


Fig. 18 Control performance of adaptive-neuro controller for the position and velocity trajectory of link 2 with inertia parameter uncertainties (20%)

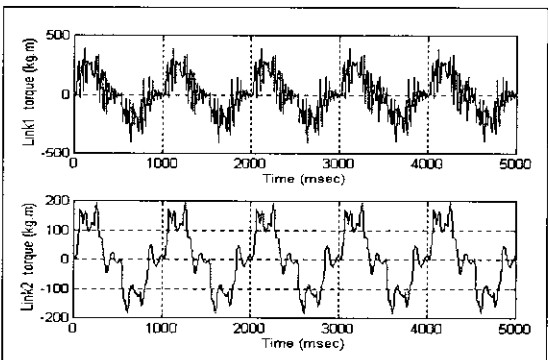


Fig. 19 Torque response for link 1 and 2 by adaptive-neuro controller with inertia parameter uncertainties (10%)