로보트 매니퓰레이터의 이산 시간 적응제어에 관한 연구

*성 관용, 이 은철, 유 재근, 남 문현

건국대학교 전기공학과

A study on the Discrete-Time Adaptive Control for Robot Manipulator

*Sung Kwan young, Lee Un Cheol, Yoo Jae guen, Nam Moon-Hyun

Dept. of Electrical Eng., Kon-Kuk Univ.

ARSTRACT

The practical implementation of model reference adaptive systems(MRAS) using digital computer requires the derivation of discrete-time adaptation laws.

This is specially important in the case of direct driver robot and light weight manipulator where inertia changes ang gravity effects are significant.

We develope a discrete-time model reference adaptive control scheme for trajectory tracking of robot manipulator.

Instead of the conventional Lyapunov approach hyperstability theory is more appealing than the Lyapunov approach.

It is better suited to discrete time systems and offers more flexibility in design by providing additional free design parameters.

Ⅰ. 서 론

최근 고도로 분업화가 진행되는 현대 산업사회에서 인간의 능력으로는 할 수 없는 일이나 하기 힘든 일이 점점 더 많아 지고 있다. 로보트는 다양한 작업을 수행하거나 동일한 작업을 반복적으로 수행할 수 있는 능력이 있기 때문에 미래 산업사회를 이끌어 갈 중요한 부분으로 인식되고 있다.

로보트의 제어방법에 있어서 로보트가 작업의 다양성 및 정밀성이 요구됨에 따라 점점 복잡한 구조를 가지게 되고 정확한 로보트 시스템의 동특성 해석과 주어진 조건에 부합되는 적절한 제어 기법의 개발이 요구되고 있다.

MRAC(Model Reference Adaptive Control)방식에서는 각 조인트에 독립적인 시불변 2차 시스템을 기준 모델로 설정하고 매니퓰레이터의 위치와 속도에 대한 이득값을 적절히 변화시켜 기준 모델을 추종한다.

지금까지의 매니퓰레이터의 적응 제어방법은 연속시간의 범주내에서 많은 연구가 이루어져 왔다.[1] 그러나 실제로 로보트는 이산시간에 바탕을 둔 디지탈 컴퓨터에 의해 제어되므로 연속시간의 정의에 기초를 둔 디지탈 신호는 샘플링시간이 크면 성능이 저하되고 폐루우프 시스템은 불안정하게 된다.

그래서 본 논문에서는 이러한 문제를 해결하기 위해 로보트 매니퓰레이터의 불규칙적인 외란에도 상관없이 정확한 궤적 추적을 위한 이산시간 모델 기준 적응 제어방법을 사용한다.

본 제어는 인코더에 의해 측정된 관절각 계측으로부터 얻은 피드백(feedback), 피드 포워드(feedforward), 외부 신호 (auxiliary signal)을 이용한다. 제어이득 행렬에 대한 적응칙을 도출하기 위해 초안정 이론(Hyperstability theory)이 이용된다. 로보트의 변수의 커다란 변동이나 부정확성에도 상관없이 궤도 추적이 가능하다.

본 논문의 모델링 방법은 앞서 발표된 일반적인 이산시간 로보트 매니 퓰레이터 모델에 기초를 두고 개발된 [2][3] 논문과는 달리 로보트 관절사이의 상호 작용을 고려하였다. 더우기 초안정 이론은 일반적인 Lyapunov 방법보다 이산시간에 더 적합하며 또한 자유로운 설계변수를 제공하므로 유연성을 한층 더 증대시킬 수 있다.

본 방식은 설계에 많은 유연성을 제공하고 또 설계자가 자유로운 설계 변수를 조절해서 제어 시스템의 성능을 개선하는 것이 가능하다. 이 이산시간 적응 알고리즘은 매우 간단하기 때문에 실시간 처리에 적합하다.

본 제어기의 성능 및 특성을 증명하기 위해 C 언어로 PUMA 560의 두번째, 세번째관절을 모델로 하여 시뮬레이션을 하였다.

Ⅱ. 이산시간 로보트 모델

이산시간의 모델을 구하기 위해서 주로 두가지의 방법이 이용되고 있다. 먼저 Neuman과 Tourassis[4]는 각각의 샘플링시간에 에너지와 운동량의 보존을 확인하는 이산시간 모델을 개발했다. 이산시간 로보트 모델을 얻기위해서는 결과적으로 각각 샘플 시간에서의 운동을 적분하여야 한다. 이 방정식은 수치적으로 반복 알고리즘을 사용하여 구할 수 는 있으나 이산 시간 모델을 얻기가 힘들다. 본 논문에서 사용되는 두번째 방법은 연속모델로 이산 시간 모델을 구현하는 것이다.

디지탈 제어기를 설계하기 위해서는 $\theta_r(k)$와 $\theta(k)$에 관계된 이산시간 로보트의 모델 방정식이 필요하다.

질량 m의 부하를 수행하는 n 관절 매니퓰레이터의 운동 방정식을 고려하자.[5]

$$B_2(\theta)\ddot{\theta}(t) + B_1(\theta,\dot{\theta})\dot{\theta}(t) + B_0(\theta)\theta(t) = u(t) \quad (1)$$

여기서 $\theta(t)$와 $u(t)$는 각각 n x 1 관절각, 관절 토크 벡터이며, $B_2(t) = B_2(\theta, \theta, m)$, $B_1(t) = B_1(\theta, \dot{\theta}, m)$, $B_0(t) = B_0(\theta, \dot{\theta}, m)$인 n x n 시변행렬이다.

2차 미분 방정식 식(1)의 이산 형태는 2차 차분 방정식으로 표현 할 수 있다.

$$A_2(k,T)\theta(k-2) + A_1(k,T)\theta(k-1) + A_0(k,T)\theta(k) = u(k) \quad (2)$$

이산 시간 모델 식(2)는 연속 모델 식(1)과 같은 로보트 특성을 나타낸다.

여기서 시간 t=kT에서 연속 시간 모델의 응답 $\theta(t)$를 찾아서 이산 시간 모델 식(2)의 응답$\theta(k)$와 같음을 보여야 한다.

식(1)과 식(2)를 표준 상태 공간 형태로 표시하면

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \quad (3)$$

$$x(k+1) = E(k,T)x(k) + F(k,T)u(k) \quad (4)$$

여기서 x(t)는 2n x 1 인 상태 벡터이고 A(t)와 B(t)는 각각 $B_2(t)$, $B_1(t)$, $B_0(t)$와 관계된 2n x 2n, 2n x n 행렬이다.

마찬가지로 E(k,T)와 F(k,T)는 각각 $A_2(k,T)$, $A_1(k,T)$, $A_0(k,T)$와 관계된 2n x 2n, 2n x n 행렬이다.

만약

$$E(k,T) = \Phi[(k-1)T, kT]$$

$$F(k,T)=\int_{kT}^{(k-1)T}\Phi[(k-1)T,\tau] \quad (5)$$

이라면 식(3)와 식(4)의 응답은 샘플링 시간 T에서 같게 나타난다.
여기서 $\Phi(\cdot,\cdot)$는 연속 시간 시스템의 천이 행렬이다.
연속 모델과 이산 모델의 행렬들 사이의 관계는 다음의 근사법을 이용하여 얻을 수 있다.

$$\theta(t) \approx \frac{1}{T}\theta(k) - \theta(k-1)$$

$$\ddot{\theta}(t) = \frac{d}{dt}\dot{\theta}(t) \approx \frac{1}{T^2}[\theta(k) - 2\theta(k-1) + \theta(k-2)] \quad (6)$$

샘플링 시간 T 가 작아질수록 근사식 (6)은 정확해진다.
식 (1)에 (6식을 대입하면 식(2)의 이득 행렬을 구할 수 있고 그 값은 다음과 같이 표시한다.

$$A_2(k,T) = \frac{B_2(k)}{T^2}, \quad A_1(k,T) = \frac{B_1(k)}{T} - \frac{2B_2(k)}{T^2}$$

$$A_0(k,T) = B_0(k) - \frac{B_1(k)}{T} - \frac{2B_2(k)}{T^2} \quad (7)$$

$$A_2(k)\theta(k-2) + A_1(k)\theta(k-1) + A_0(k)\theta(k) = u(k) \quad (8)$$

식(8)은 로보트운동 동안의 이산시간 간격에서의 일반적인 로보트 동작을 나타내고 있다. 물리적 시스템의 수학적 모델로써 이 방정식은 실제 로보트 동작의 근사식이다.

Ⅲ. 적응제어기의 설계

여기에서는 로보트 모델 식 (8)에 의해서 부하의 변화와 불확실한 로보트 변수에도 불구하고 로보트 관절각 벡터 $\theta(k)$가 기준 궤도 벡터 $\theta_r(k)$를 추종하는 이산 시간 적응제어기의 설계를 유도한다. 적응 제어의 블러 선도는 그림 1에서 나타낸다.

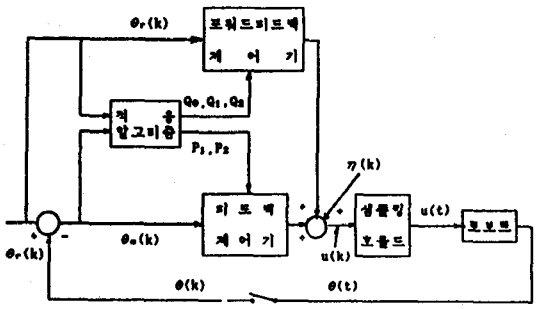

그림1. 이산시간 적응제어 블록 선도

n x 1 관절각 오차 벡터를

$$\theta_e(k) = \theta_r(k) - \theta(k)$$

라 하자. $\theta_e(k) = \theta_r(k) - \theta(k)$을 식 (8)에 대입하면

$$A_0(k)\theta_e(k) = u(k) - A_1(k)\theta_e(k-1) - A_2(k)\theta_e(k-2) + A_0(k)\theta_r(k) + A_1(k)\theta_r(k-1) + A_2(k)\theta_r(k-2) \quad (9)$$

과 같은 관절각 오차 방정식을 얻을 수 있다.
식(8)는 관절각 오차 $\theta_e(k)$를 제어하기위해서 다음과 같은 제어식을 이용한다.

$$u(k) = P_1(k)\theta_e(k-1) + P_2(k)\theta_e(k-2) + Q_0(k)\theta_r(k) + Q_1(k)\theta_r(k-1) + Q_2(k)\theta_r(k-2) + \eta(k) \quad (10)$$

여기서 $P_1(k)$, P_2는 관절각 오차에 작용하는 시변 피드백 행렬이고 Q_0, Q_1, Q_2는 기준궤적에 작용하는 시변 피드포워드 행렬이다. 보조 신호 $\eta(k)$는 적응과정을 항상 시키기위해 이용한다.
식(10)을 식(9)에 대입하면 페루우프 시스템에 대한 관절각 오차 방정식을 얻는다.

$$A_0(k)\theta_e(k)+[P_1(k)+A_1(k)]\theta_e(k-1)+[P_2(k)+A_2(k)]\theta e(k-2) = [A_0(k) - Q_0(k)]\theta_r(k) + [A_1(k) - Q_1(k)]\theta_r(k-1) + [A_2(k) - Q_2(k)]\theta_r(k-2) - \eta(k) \quad (11)$$

매니플레이터에 요구되는 이상적인 기준 모델식은 다음과 같다.

$$\theta_{em}(k) + C_1\theta_{em}(k-1) + C_2\theta_{em}(k-2)=0 \quad (12)$$

여기서 $\theta_{em}(k)$는 기준 모델의 n x 1 관절각 오차 벡터이고 C_1, C_2는 관절각 오차와 결합되어 시간에 따라 감소되도록 선택된 n x n 상수 행렬이다.
모델기준 적응제어의 용어에 있어서 식(11)와 식(12)은 각각 적응 시스템과 기준 모델로 표시된다. 관절 오차의 디커플링에 대해 $C_1=diag\{c_{i1}\}$ 와 $C_2 = diag\{c_{2i}\}$ $i = 1,2,\ldots,n$을 선택한다. 오차가 0으로 감소하기 위해 기준 모델 식(12)의 특성 다항식 $\Delta(z)$의 근 $\lambda_{i1}, \lambda_{2i}$은 복소수 z평면의 단위원 안에 놓여야 한다. 이 식은 다음과 같이 표현된다.

$$\Delta(z) = \det(I_n z^2 + C_1 z + C_2) = \prod_{i=1}^{n}\delta_i(z)$$

$$\delta_i(z) = z^2 + c_{i1}z + c_{2i}) = (z + \lambda_{i1})(z + \lambda_{2i})$$

이 때 행렬 C_1 과 C_2의 대각 요소는

$$c_{i1} = \lambda_{i1} + \lambda_{2i}\ ; \qquad c_{2i} = \lambda_{i1}\lambda_{2i}$$

이고 여기서 기준 모델의 안정성을 위해서 $|\lambda_{i1}| < 1$, $|\lambda_{2i}| < 1$ 을 만족해야 한다.
식(12)의 해는

$$\theta_{em}(k) = \Phi_m(k)\theta_{em}(0) \quad (13)$$

여기서 $\Phi_m(k)$는 기준 모델 식(12)의 천이 행렬이고 $\theta_{em}(0)$는 기준 모델의 초기값이다. 만약 $\theta_{em}(0)$가 0으로 선택되면 $\theta_{em}(k)$는 분명히 0이다. 즉 모든 $k \geq 0$ 에 대한 기준 모델의 안정도에 의해 $\theta_{em}(k) \simeq 0$ 이 된다.
식(11)에 좌우되는 로보트 관절각 $\theta(k)$이 $\theta_{em}(k) \simeq 0$ 인 기준 모델 동력학 식(12)에 도달하도록 적응 알고리즘을 고안하는 것이 목적이다. 이런 목적을 달성하기위해서 이상(ideal)과 실제 오차의 차를 정의한다.

$$\varepsilon(k)= \theta_{em}(k) - \theta_e(k) \quad (14)$$

식(11), 식(12)과 식(14)을 결합하면 아래와 같다.

$$\varepsilon(k) + C_1\varepsilon(k-1) + C_2\varepsilon(k-2)+w(k) = 0 \quad (15a)$$

$$w(k) = [C_1 - A_0^{-1}(A_1(k) + P_1(k))]\theta_e(k-1) + [C_2 + A_0^{-1}(A_2(k) + P_2(k))]\theta_e(k-2) + A_0[A_0^{-1}(k)- Q_0(k)]\theta_r(k) + A_0^{-1}[A_1(k)- Q_1(k)]\theta_r(k-1) + A_0^{-1}[A_2(k)- Q_2(k)]\theta_r(k-2) - A_0^{-1}\eta(k)$$
$$= w_1(k) + w_2(k) + \cdots + w_6(k) \quad (15b)$$

적응제어 알고리즘은 적응오차 동력학 식(15)가 안정한 피드백 이득행렬 $P_1(k)$,$P_2(k)$, 피드포워드 이득행렬 $Q_0(k)$, $Q_1(k)$, $Q_2(k)$와 보조 신호 $\eta(k)$를 구하는 것이다.
즉 식(14)의 $\varepsilon(k)$가 점진적으로 0에 도달하면 관절각 오차 벡터 $\theta_e(k)$는 $\theta_e(k) \simeq 0$ 을 의미하는 기준 모델 오차 벡터 $\theta_{em}(k) \simeq 0$ 와 같게 된다. 그러므로 $\theta(k) = \theta_r(k)$로 궤적 추적이 가능하다.

$$\varepsilon(k) = \theta_{em}(k) - \theta_e(k) = \theta_{em}(k) - \theta_e(k)$$

$$\begin{bmatrix} \varepsilon(k-1 \\ \varepsilon(k) \end{bmatrix} = \begin{bmatrix} 0 & I_n \\ -C_2 & -C_1 \end{bmatrix} \begin{bmatrix} \varepsilon(k-2) \\ \varepsilon(k-1) \end{bmatrix} - \begin{bmatrix} 0 \\ I_n \end{bmatrix} w(k) \quad (16)$$

적응 제어 알고리즘을 전개하면 다음과 같다.

$$\upsilon(k) = D\begin{bmatrix} \varepsilon(k-1) \\ \varepsilon(k) \end{bmatrix} \quad (17)$$

$$\begin{aligned} w(k) = {} & \psi_1(v, \theta_*)\theta_*(k-1) + \psi_2(v, \theta_*)\theta_*(k-2) + \\ & \psi_3(v, \theta_r)\theta_r(k) + \psi_4(v, \theta_r, \theta_r(k-1) + \\ & \psi_5(v, \theta_r)\theta_r(k-1) + \rho(v) \end{aligned} \quad (18)$$

여기서 $\upsilon(k)$은 n x 1 벡터이고 D 는 n x 2n상수 벡터이다. $\psi_1(v, \theta_*), \ldots, \psi_5(v, \theta_r)$은 n x 2n 행렬이고 $\rho(v)$은 n x 1벡터이다.
식(16)부터 식(18)까지의 식에 의해 표시된 적응 동력학이 안정하고 적응에러가 점진적으로 0에 접근하는 것을 확인하기 위해 popov의 초안정 이론(hyperstability)을 사용한다. 이 이론은 적응 과정의 동력학 방정식이 피드백 형태로 배열되는 것을 필요하다.
다음 두가지 조건을 만족하면 적응 알고리즘 식(16)-식(18)은

$$\lim_{k\to\infty} \begin{bmatrix} \varepsilon(k-1) \\ \varepsilon(k) \end{bmatrix} = 0$$

가 되므로 안정하다.
조건 1: 포워드 블럭 $H(z) = zD(zI_{2n} - C)^{-1}B$ 의 전달함수행렬은 SPR(Strictly Positive Real)이다.
여기서 $C = \begin{bmatrix} 0 & 2n \\ -C_2 & -C_1 \end{bmatrix}$ $B = \begin{bmatrix} 0 \\ I_n \end{bmatrix}$
이다.

조건 2: 피드백 블럭의 입출력이 모든 k_1에대해

$$\sum_{k=0}^{k_1} \nu^T(k)w(k) \geq -\gamma^2 \text{ 을 만족한다.}$$

위의 두가지 조건을 만족하면 제어기 이득은 다음과 같이 된다.

$$\hat{P}_1(k) = \hat{P}_1(k-1) + \hat{\theta}_*(k)\hat{\theta}_*^T(k-1)E_{1P} + \hat{\theta}_*(k-1)\hat{\theta}_*^T(k-2)[E_{11} - E_{1P}]$$

$$\hat{P}_2(k) = \hat{P}_2(k-1) + \hat{\theta}_*(k)\hat{\theta}_*^T(k-2)E_{2P} + \hat{\theta}_*(k-1)\hat{\theta}_*^T(k-3)[E_{21} - E_{2P}]$$

$$\hat{Q}_0(k) = \hat{Q}_0(k-1) + \hat{\theta}_*(k)\hat{\theta}_r^T(k)F_{0P} + \hat{\theta}_*(k-1)\hat{\theta}_r^T(k-1)[F_{01} - F_{0P}]$$

$$\hat{Q}_1(k) = \hat{Q}_1(k-1) + \hat{\theta}_*(k)\hat{\theta}_r^T(k-1)F_{1P} + \hat{\theta}_*(k-1)\hat{\theta}_r^T(k-2)[E_{11} - E_{1P}]$$

$$\hat{Q}_2(k) = \hat{Q}_2(k-1) + \hat{\theta}_*(k)\hat{\theta}_r^T(k-2)E_{2P} + \hat{\theta}_*(k-1)\hat{\theta}_r^T(k-3)[E_{21} - E_{2P}]$$

$$\eta(k) = \beta_P\hat{\theta}_*(k) + (\beta_1 - \beta_P)\hat{\theta}_*(k-1) \quad (19)$$

여기서 $\hat{\theta}_*(k) = R_2\hat{\theta}_*(k-1) + R_3\hat{\theta}_*(k)$ (20a)

위 방정식에서 $E_{0P}, E_{01}, \cdots, F_{2P}, F_{21}$ 은 상수 SPD 적응이득 행렬이고, β_1와 β_P은 양의 스칼라이고 밑첨자 p 와 1은 각각 비례, 적분부분을 표시한다.
더우기 R_2와 R_3은 대각요소 r_{21}와 r_{31}가 다음과 같은 n x n 대각행렬이다.

$$r_{21} = \alpha_1\lambda_{11}\lambda_{21} \quad (\lambda_{11} + \lambda_{21}) \quad (20b)$$

$$r_{31} = \alpha_1(1 + \lambda_{11}\lambda_{21}) \quad (20c)$$

여기서 α_1는 양의 상수이고 $\lambda_{11}, \lambda_{21}$은 앞에서 설명한 $|\lambda_{11}| < 1$, $|\lambda_{21}| < 1$ 사이에 있는 오차 기준 모델의 기준치이다. 피드백 이득 $P_1(k)$, $P_2(k)$은 관절 오차 벡터만 종속하는 반면에 포워드 이득 $Q_0(k)$, $Q_1(k)$, $Q_2(k)$은 관절각 벡터와 기준 궤도 오차의 양쪽 모두에 종속적이다.
방정식 식(10), 식(19) 식(20)은 적응제어의 알고리즘을 구성한다. SPD 행렬 $E_{0P}, E_{11}, \cdots, F_{2P}, F_{21}$과 $\beta_P, \beta_1, \alpha_1$과 고유치 $\lambda_{11}, \lambda_{21}$은 설계자에 의해 지정되어야 한다.

특히 $\hat{\theta}$에 대한 간단한 표현은 이것은 소위 "Dead-beat 제어"라고 한다. 이 경우에 식 (19a)는 다음과 같이 간단해 진다.

$$\hat{\theta}_*(k) = R_3\theta_*(k); \quad R_3 = \mathrm{diag}\{\alpha_1\} \quad (21)$$

행렬 $E_{0P}, E_{11}, \cdots, F_{2P}, F_{21}$과 스칼라 α_1의 요소의 큰 값은 더 높은 적응이득을 갖고 오차를 빠르게 감소시킨다.
식(10), 식(19), 식(20)에 의해 주어진 적응제어 법칙은 매우 간단하고 실시간 제어가 적당하다. $P_1(k), \cdots, Q_2(k)$은 비슷한 구조를 갖으므로 식(19)을 계산하기위해 같은 서브루틴을 불러 계산할 수 있으므로 간략화시킬 수 있다.

V. 시뮬레이션

여기에서는 앞에서 유도한 적응제어 법칙을 적용하여 PUMA560의 제 2, 3관절의 매니퓰레이터의 변수를 이용하여 시뮬레이션하였다.
부하가 걸리지 않았을 때의 로보트의 운동 방정식은 아래와 같다.

$$M(\theta)\ddot{\theta} + N(\theta, \dot{\theta}) + G(\theta) + H(\dot{\theta}) = u(t) \quad (22)$$

로보트가 질량 m의 부하가 걸릴 때 끝점(end-effector)는 중력 g에 의한 가속도를 가지고 움직이는 $f(t) = m(\ddot{x}(t) + g)$의 힘을 요구한다. 여기서 $J(\theta)$는 자코비안 행렬로서 끝점 속도 $\dot{x}$ 와 $\dot{x}(t) = J(\theta)\dot{\theta}(t)$에 대응되는 관절 속도 $\theta(t)$와 관계있다. 그러므로 동력학 방정식에서 요구된 가속도 $\ddot{x}$는 $\ddot{x}(t) = J(\theta, \dot{\theta}))\dot{\theta}(t) + J(\theta)\ddot{\theta}(t)$이다.
여기서 식 (22)에서 부하를 움직이는데 요구되는 토오크를 가산하면 아래식으로 표현된다.

$$[M(\theta) + mJ^T(\theta)J(\theta)]\ddot{\theta}(t) + [mJ^T(\theta)\dot{J}(\theta, \dot{\theta})]\dot{\theta}(t) +$$

$$N(\theta, \dot{\theta}) + H(\dot{\theta}) + G(\theta) + mJ^T(\theta) = u(t) \quad (23)$$

2개의 링크를 직접 구동 매니퓰레이터에 대해서 식(23)의 형태로 적용하면 다음과 같다.[7]
θ_1은 수평선과 링크 1 사이의 각으로 정의되고 θ_2는 링크1과 링크 2사이의 각이다. 변수들은 $a_1 = 49.3$, $a_2 = 0.94$, $a_3 = 0.90$, $a_4 = 68.95$, $a_5 = 10.64$, $b_1 = 6.82$Nt.m.sec/rad, $b_1 = b_4 = 3.5$Nt.m.sec/rad이다. 와 PUMA 560의 제2.3 관절에 적용 시킨다. 이들 변수들은 PUMA 560의 링크 2,3과 유사한 무게와 길이를 지닌 매니퓰레이터의 변수이다. 부하는 t = 1초에 0에서 1, 3, 5kg으로 변한다.
기준 궤적 $\theta_{ri}(t)$, i = 1,2 는

$$\theta_{ri}(t) = \begin{cases} \theta_i(0) + (\theta_i(\infty))\left[\dfrac{t}{\tau_i} - \dfrac{1}{2\pi}\sin\dfrac{2\pi t}{\tau_i}\right] & 0 \leq t \leq \tau_i \\ \theta_i(\infty) & t \geq \tau_i \end{cases} \quad (24)$$

여기서 $\theta_i(0)$와 $\theta_i(\infty)$는 각각 기준 궤적의 초기치와 최종치이고 τ_i는 초기 위치에서 최종 위치까지 움직이는데 요구되는 시간이다. 기준 궤적 식(24)를 이용하는 이유는 연속적인 속도와 가속도를 구하기 위해서이다. 위의 $\theta_{ri}(t)$은 $\theta_{ri}(k)$을 구하기위해 샘플링한다. 제어기의 실행을 설명하기 위해서 초기위치는 "arm-up" 위치에 해당하는 $\theta_1(0) = 90$, $\theta_2(0) = 0$, 최종위치는 $\theta_1(\infty) = 90$, $\theta_2(\infty) = 0$ 이다.

$$u(k) = \hat{P}_1(k)\hat{\theta}_*(k-1) + \hat{P}_2(k)\hat{\theta}_*(k-2) + \eta(k) \quad (25)$$

$$\hat{P}_1(k) = \hat{P}_1(k-1) + 30\hat{\theta}_*(k-1)\hat{\theta}_*^T(k-2) \quad (25a)$$

$$\hat{P}_2(k) = \hat{P}_2(k-1) + 30\hat{\theta}_*(k-1)\hat{\theta}_*^T(k-3) \quad (25b)$$

$$\eta(k) = \eta(k-1) + 30\hat{\theta}_*(k) - 29.98\hat{\theta}_*(k-1) \quad (25c)$$

$$\hat{\theta}_*(k) = -137.2\hat{\theta}_*(k-1) + 149\hat{\theta}_*(k) \quad (25d)$$

이득과 보조 신호의 초기치는 $\hat{P}_1(0)$, $\hat{P}_2(0)$, $\eta(0)$는 임의적으로 0으로 한다.
그림2는 관절1의 기준 궤적과 실제 궤적를 보여 주고 있으며 그림 3는 관절2의 기준 궤적과 실제 궤적를 나타낸다. 그림 2와 3의 점선은 기준 궤적를 나타내고 실선은 실제 궤적를 나타내고 있으나 오차가 0.8도를 넘지 않으므로 T =3에 근접해서야 점선이 보이고 있다. 그림 4과 5는 관절 1과 관절 2의 기준 궤적과 실제 궤적 사이의 오차를 나타낸다. 최대 오차각은 그림과 같이 1초후 최대 5kg의 하중이 걸림에도

불구하고 0.8도를 넘지 않고 있다. 그림 6과 7는 관절 1과 관절 2의 토오크를 보여준다. 위의 모든 과정은 샘플링 시간 T = 0.01sec로 하여 실행하였다. 위의 모든 시뮬레이션 프로그램은 C 언어를 사용하였다.

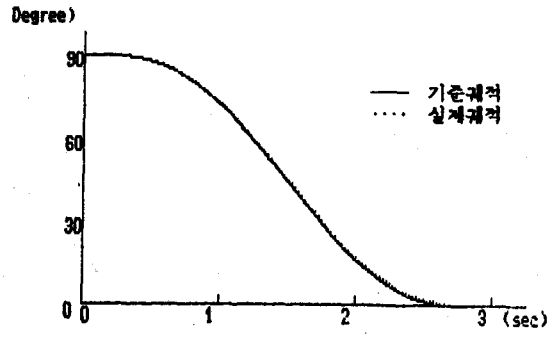

그림2. 관절 1의 기준궤적과 실제궤적

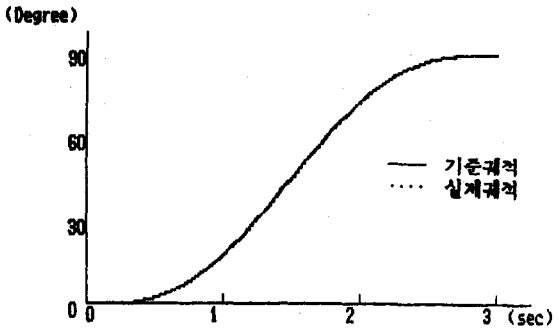

그림3. 관절 2의 기준궤적과 실제궤적

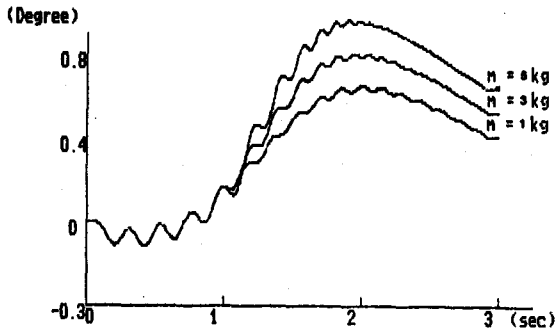

그림4. 관절 1의 오차 관절각

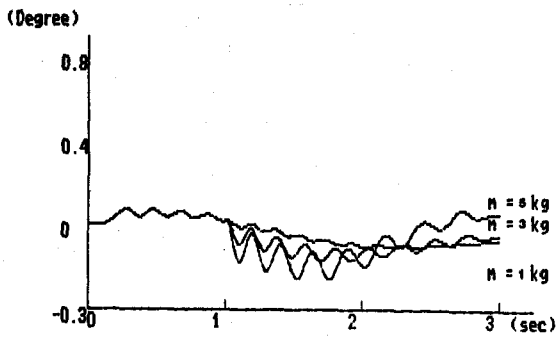

그림5. 관절 2의 오차 관절각

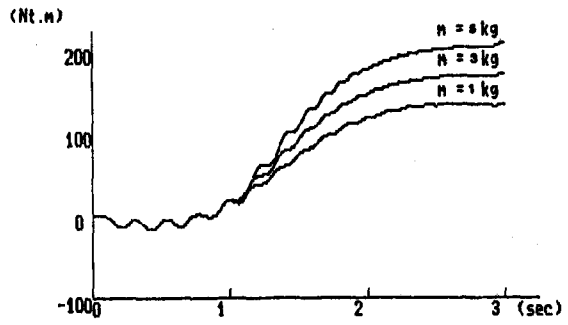

그림6. 관절 1의 토오크

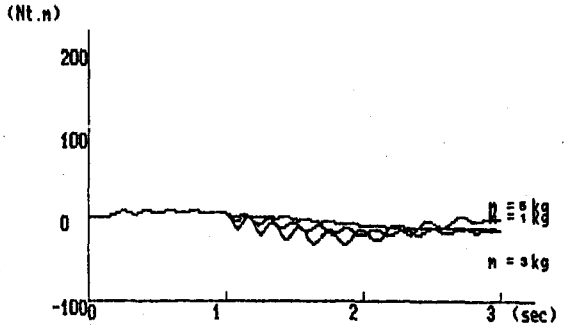

그림7. 관절 2의 토오크

Ⅵ. 결 론

본 논문에서는 관절 위치의 측정과 기준 궤적을 이용하여 이산 시간 적응 알고리즘을 설계하였다. 그리고 부하와 로보트 특성은 불확실하며, 적응법칙은 로보트 변수를 거의 알지못하는 상황에서도 점진적으로 궤적를 추종하는 초안정 이론을 이용했다.

위의 실험에서 1초후 질량 1, 3, 5kg을 부가했음에도 불구하고, 매우 정확하게 기준 궤도를 추정했다. 이와 같이 로보트 특성을 알지 못해도 이산시간 적응 알고리즘을 사용하여 정확한 궤적 추종이 가능하다. 그리고 이 알고리즘은 많은 유연성을 제공하며 또한 설계자가 자유로운 설계 변수를 조절해서 제어 시스템의 성능을 개선하는 것이 가능하다. 이산 시간 적응제어 알고리즘은 매우 간단하므로 계산적으로도 빠르고, 로보트 매니플레이터의 실시간 디지탈 제어에 적당하다. 로보트 매니플레이터 모델에 사용한 컴퓨터 시뮬레이션 연구는 이산 시간 적응 알고리즘이 불확실한 부하변동과 로보트 변수들에도 불구하고 만족스럽게 작동됨을 확인할 수 있었다.

참고 문헌

[1] K. Y. Lim and M. Eslami,"Adaptive Controller Design for Robot Manipulators Systems using Lyapunov Direct Method",IEEE Trans.Auto.Control,30,pp.1229-1233,1985.

[2] A. J. Koivo and T. H. Guo,"Adaptive Linear Controller for Robotic Manipulators", IEEE Trans. Auto. Control, 28,pp.162-170, 1983.

[3] C. S. G. Lee and M. J. Chung," An Adaptive Control Strategy for Mechanical Manipulators", IEEE Trans. Auto. Control, 29,pp. 837-840, 1984.

[4] C. P.Neumann and V.P. Tourassis,"Discrete - Dynamic Robot Models" IEEE Trans. Syst. Man. Cybern.,SMC-15, pp.193-304, 1985.

[5] A.Balestrino, G.DeMariaand L.Sciavicco,"An Adaptive Model Following Control for Robot Manipulators",ASME Journal of Dyanamic Systems Measurement and Control, 105,pp,143-150, 1983.

[6] Y. D.Landaw,Adaptive Control-Model Reference Aproach Marcel Dekker, New York, 1979.

[7] M.tarokh, "A Discrete-Time Adaptive Control Scheme for Robot Manipulator", Journal of Robotics Systems, 1990.