

論 文

컨베이어 추적을 위한 로봇 팔의 강인한
적응 제어계 설계

正會員 嚴 基 煥* 正會員 孫 東 高* 正會員 金 主 홍**

A Design of Robust Adaptive Control Systems
of Robot Arms for Conveyor Tracking

Ki Hwan EOM*, Dong Seol SON*, Ju Hong KIM* *Regular Members*

要 約 로봇 팔이 컨베이어를 추적하는 강인한 적응제어계를 작업좌표에서 설계하였다. 제안한 설계방식은 모델계법형 설계방식과 Popov의 초안정정리를 기본으로 하였고 보조입력을 사용하여 Lyapunov 함수 $V(t)$ 와 시간미분 \dot{V} 와의 비의 크기를 크게하여 적응성을 좋게하여 과도특성을 개선하였다.
설계변수 λ 와 하중함수 L_k 에서 L_k 의 값이 λ 의 값보다 작은 경우에는 계통이 안정하나 L_k 의 값이 λ 의 값보다 큰 경우에는 계통이 불안정하였다.

ABSTRACT In this paper, we presents a robust adaptive control system design method in the work coordinate of the robot arm for conveyor tracking.

In the design, if the weighting function L_k is smaller than the design parameter λ , then the transient characteristics of system becomes stable, if L_k is larger than λ , then the system becomes unstable.

Proposed design method presented here is based on model referenece adaptive control and Popov stability theorem.

By the utilization of an auxilary input, it is improved the transient characteristics of the system in comparison with the conventional model reference adaptive control, since the rate of \dot{V} and $V(t)$ is large.

The usefulness of a proposed design method has been confirmed by computer simulations.

I. 서 론

공장 자동화에서 많은 작업들이 로봇과 컨베이어의 결합으로 행하여 지는 경우가 많다. 즉 컨베이어상의 작업물이 로봇의 작업 공간내에 이동되면 로봇이 작업을 종료한 후 다음의

* 柳韓工業專門大學 電氣 電子科
Dept. of Electrical Yu Han Technical College.
** 東國大學校 電子工學科
Dept. of Electronics Engineering Dongguk University
論文番號 : 90-95(接受1990. 10. 13)

작업단계로 이동된다. 이 과정에서 작업물이 로봇의 작업공간내로 이동된 후 컨베이어를 정지시킨 상태에서 작업을 하는 방식은 효율이 떨어지므로 로봇이 컨베이어상의 작업물을 추적하여 작업을 연속적으로 시행하면 작업 효과를 높일 수 있다.

이러한 작업을 하기 위하여는 로봇과 컨베이어와의 상대위치, 상대속도가 0으로 유지되어야 한다. 즉 로봇의 팔이 컨베이어를 정확히 추적하는 제어를 설계하지 않으면 안된다. 그러나 로봇 팔은 관절부에 존재하는 원심력, 코리올리력, 점성마찰등의 비선형성, 시가변성, 불확실성등에 의해 정확한 동적 모델링이 어려우므로 고도의 동적학적 성능을 가진 제어기의 설계는 쉽지 않다.

그러므로 로봇 팔의 동적 모델링에 있어서 매개변수가 비선형성과 불확실성등을 가진 경우에서도 좋은 제어성능을 가지고, 부하의 변동에 모델링 오차에 대해서는 잘 적응하는 제어방식을 가진 로봇의 설계와 제자에 관한 많은 연구가 진행되고 있다^{1) 2) 3)}.

적응제어는 1950년대 후반에 항공기의 오토 파일럿 (Auto Pilot)의 설계를 시작으로 그 후 이론과 하드웨어의 발달에 따라 1970년대 전반에는 임출력 신호와의 정보로 제어가 가능한 제어기의 설계방식이 확립되었다^{4) 5)}.

적응제어의 이론은 Lyapunov의 안정이론 또는 Popov의 초안정정리등의 안정이론과 正實(positive real) 개념을 기본으로 한 모델 규범형 적응제어제 (Model Reference Adaptive Control System ; MRACS)의 설계법이 있고 또 하나는 확률적 제어이론과 同定 (identification) 理論을 기본으로 하는 자기 동조 레귤레이터 (Self Tuning Regulator ; STR) 또는 자기 동조 컨트롤러 (Self Tuning Controller ; STC)의 설계법이 있다^{6) 2) 3) 7)}.

최근 이 적응제어 이론의 로봇제어에 관한 연구는 매우 활발하다^{8) 9)}.

여기서는 작업좌표계에서 미지의 매개변수를 갖는 로봇 팔의 컨베이어 추적이 보증되는

강인한 적응제어계를 설계한다. 설계방식은 MRACS 를 기본으로 Lyapunov 함수 V(t)와 시간미분 \dot{V} 의 비를 크게 하기 위하여 보조입력을 넣어 과도특성을 개선하고, 오차방식으로부터 強正實 (strictly positive real)한 선형 부분을 분리하여 Popov의 초안정정리를 적용하여 강인한 적응제어계를 설계한다.

또한 설계변수와 하중함수의 관계에서 제어기의 과도특성을 조사하고 제안한 강인한 적응제어제와 일반적인 MRACS를 컴퓨터 시뮬레이션에 의하여 비교 검토하고 그 유효성을 확인한다.

II. 로봇의 동적모델

일반적으로 n관절 로봇 팔의 운동방정식은

$$H(\theta)\ddot{\theta} + C(\dot{\theta}, \theta) + D\dot{\theta} + G(\theta) = \tau \quad (1)$$

의 형식으로 된다¹⁰⁾. 여기서 $H(\theta) \in R^{n \times n}$ 은 관성행렬, $C(\dot{\theta}, \theta) \in R^n$ 은 코리올리력, 원심력 벡터, $D \in R^{n \times n}$ 은 점성마찰행렬, $G(\theta) \in R^n$ 은 중력벡터, $\tau \in R^n$ 은 입력벡터, $\theta, \dot{\theta}, \ddot{\theta} \in R^n$ 은 관절각, 관절각속도, 관절각 가속도이다.

컨베이어를 추적하는 로봇 팔의 제어계를 로봇 팔에 고정된 작업좌표계를 이용하면, 컨베이어로 이동되는 작업물의 위치와 속도는 n차원의 작업좌표 X와 관절좌표 θ 와의 관계에서

$$x = f(\theta) \quad (2 a)$$

$$\dot{x} = J(\theta)\dot{\theta} \quad (2 b)$$

$$\ddot{x} = \dot{J}(\theta)\dot{\theta} + J(\theta)\ddot{\theta} \quad (2 c)$$

이 성립하고 여기서 $j(\theta) = \partial f(\theta) / \partial \theta$ 는 $f(\theta)$ 의 야코비행렬이다.

식(2)를 이용하여 로봇 팔의 운동방정식을 구하면

$$H(\theta)J_1\ddot{x} - H(\theta)J_2\ddot{x} + C(\dot{\theta}, \theta) + DJ_1\dot{x} + G(\theta) = \tau \quad (3)$$

이다. 여기서 $J_1 = J^{-1}(\theta)$, $J_2 = J^{-1}(\theta)J(\theta)J^{-1}(\theta)$ 이다.

식(3)의 운동방정식의 각항은 기본변수와 기지의 비선형행렬의 곱의 합으로 쓸 수가 있다. 즉,

$$\mathcal{L} = (\mathcal{L}_1, \mathcal{L}_2, \dots, \mathcal{L}_q) \\ = (h_x, h_y, h_z, \dots, r \dots D_q) \quad (4)$$

단, h 는 관성모우멘트, D 는 회전축에 점성마찰계수, r 은 ma 로서 m 은 링크의 질량, a 는 회전축으로부터 링크의 重心位置까지의 거리로서, 이를 기본변수로 하고

$$H(\theta) = \sum_{k=1}^m \alpha_k H_k(\theta) \quad (5-a)$$

$$C(\theta, \theta) = \sum_{k=1}^m \alpha_k C_k(\dot{\theta}, \theta) \quad (5-b)$$

$$G(\theta) = \sum_{k=1}^{m'} \alpha_k G_k(\theta) \quad (m \leq m') \quad (5-c)$$

$$D = \sum_{k=n+1}^q \alpha_k D_k(\theta) \quad (5-d)$$

로 표시되고, 여기서 $H_k(\theta)$, $C_k(\dot{\theta}, \theta)$, $G_k(\theta)$, D_k 는 각각 적당한 크기의 행렬로 로봇의 구조에 의하여 정하여 진다.

따라서 식(3)에 식(5)를 대입하면 다음과 같이 된다.

$$H(\theta)J_1\ddot{x} - H(\theta)J_2\ddot{x} + C(\dot{\theta}, \theta) + DJ_1\dot{x} + G(\theta) \\ = \sum_{k=1}^m \alpha_k H_k(\theta)J_1\ddot{x} - \sum_{k=1}^m \alpha_k H_k(\theta)J_2\ddot{x} \\ + \sum_{k=1}^m \alpha_k C_k(\dot{\theta}, \theta) + \sum_{k=1}^m \alpha_k G_k(\theta) \\ + \sum_{k=n+1}^q \alpha_k D_k J_1 \dot{x} \\ \triangleq \sum_{k=1}^m \alpha_k H^*_k(\theta)\ddot{x} + \sum_{k=1}^q \alpha_k B_k(\dot{\theta}, \theta) \quad (6)$$

$$\text{단, } H^*_k(\theta) = H_k(\theta)J_1 \quad (7-a)$$

$$B_k(\dot{\theta}, \theta) = -H_k(\theta)J_2\ddot{x} + C_k(\dot{\theta}, \theta) + G_k(\theta) \quad (k=1, 2, \dots, m) \quad (7-b)$$

$$B_k(\dot{\theta}, \theta) = G_k(\theta) \quad (k=m+1, \dots, m') \quad (7-c)$$

$$B_k(\dot{\theta}, \theta) = D_k J_1 \dot{x} \quad (k=m'+1, \dots, q) \quad (7-d)$$

이다.

식(6)을 아래와 같이 입력 τ 를 매개변수에 선형성을 유지하며 다음과 같이 변화한다.

$$\tau = -\lambda\ddot{x} + u \quad (8)$$

여기서 λ 는 설계변수로서 입력의 有界性이 보증되도록 결정되며 u 는 새로운 입력이다.

식(8)을 식(6)에 대입하면,

$$\ddot{x} = -\frac{\sum_{k=1}^m \alpha_k H^*_k(\theta)\ddot{x}}{\lambda} - \frac{\sum_{k=1}^q \alpha_k B_k(\dot{\theta}, \theta)}{\lambda} + \frac{u}{\lambda} \quad (9)$$

와 같으며 식(9)는

$$\dot{x} = A_1 x + \sum_{k=1}^q A_{2k} g_k(x, t) + B U \quad (10)$$

로 된다. 단,

$$A_1 = \begin{bmatrix} 0, I_n \\ 0, 0 \end{bmatrix} \quad (2n \times 2n) \quad (11-a)$$

$$A_{2k} = \begin{bmatrix} 0 \\ \mathcal{L}_k I_n \end{bmatrix} \quad (2n \times n) \quad (k=1, 2, \dots, q) \quad (11-b)$$

$$B = \begin{bmatrix} 0 \\ I/\lambda, I_n \end{bmatrix} \quad (2n \times n) \quad (11-c)$$

$$g_k = -\frac{-\sum_{k=1}^m H^*_k(\theta)\ddot{x} - \sum_{k=1}^q B_k(\dot{\theta}, \theta)}{\lambda} \quad (11-d)$$

이다.

이것에 대하여 추적하는 컨베이어에 대한 정보는 미리 아는 n 입력 $2n$ 자원 선형계로 생각하여 다음과 같은 규범모델로 정한다.

$$\begin{aligned} \dot{x}_m &= \begin{bmatrix} 0, & I_n \\ -k_1 I_n, & -k_2 I_n \end{bmatrix} x_m + \begin{bmatrix} 0 \\ I_n \end{bmatrix} U_m \\ &= A_m X_m + B_m U_m \end{aligned} \quad (12)$$

III. 모델정합을 위한 조건

식(10)의 제어대상과 식(12)의 규범모델의 모델정합 조건은 다음과 같이 알려져 있다^[2].

<조건1> $A_m - A_1 - B_m C_{a1}$ 을 만족하는 $n \times 2n$ 의 정수행렬 C_{a1} 이 존재한다.

<조건2> $A_{2k} = B_m C_{a2k}$ 를 만족하는 $n \times n$ 의 정수행렬 C_{a2k} 가 존재한다.

<조건3> $B = B_m C_b$ 를 만족하는 $n \times n$ 의 유계정수행렬 C_b 가 존재한다.

<조건4> C_b 는 정정행렬이다.

여기서는

$$C_{a1} = [-k_1 I_n, -k_2 I_n] \quad (13 \text{ a})$$

$$C_{a2k} = \alpha_k I_n \quad (k=1, 2, \dots, q) \quad (13 \text{ b})$$

$$C_b = 1/\lambda I_n \quad (13 \text{ c})$$

로 놓으면

$$A_m - A_1 = B_m C_{a1} \quad (14 \text{ a})$$

$$A_{2k} = B_m C_{a2k} \quad (14 \text{ b})$$

$$B = B_m C_b \quad (14 \text{ c})$$

로 되어 모델 정합의 조건을 만족하고 있는 것을 알 수 있다.

IV. 강인한 적응제어계 설계

제어입력 U 를

$$U = -F_1(x, t)x - \sum_{k=1}^q F_{2k}(x, t)g_k + F_3(x, t)U_m \quad (15)$$

와 같이 정하고 여기서 $F_1 \in R^{n \times 2n}$, $F_{2k} \in R^{n \times n}$, $F_3 \in R^{n \times n}$ 은 이득행렬이다. 식(15)를 식(10)에 대입하고 모델정합의 조건을 사용하여 오차방정식을 도출하면

$$\begin{aligned} \dot{e} &= \dot{x}_m - \dot{x} \\ &= A_m e + B_m [C_{a1} + C_b F_1(x, t)]x - \sum_{k=1}^q B_m [C_{a2k} - C_b F_{2k}(x, t)]g_k(x, t) \\ &\quad + B_m [I - C_b F_3(x, t)]U_m \end{aligned} \quad (16)$$

이 된다.

여기서 $F_1(x, t)$, $F_{2k}(x, t)$ 를 조정한다면 규범모델에의 추적이 가능하다. 그러나 본 제어계에

$$\begin{aligned} F_1(x, t) &= -C_b^{-1} C_{a1} \\ &= -\lambda [-k_1 I_n, -k_2 I_n] \end{aligned} \quad (17 \text{ a})$$

$$\begin{aligned} F_3(x, t) &= C_b^{-1} \\ &= \lambda I_n \end{aligned} \quad (17 \text{ b})$$

$$\begin{aligned} F_{2k}(x, t) &= C_b^{-1} C_{a2k} \\ &= \lambda \alpha_k I_n \end{aligned} \quad (17 \text{ c})$$

이므로 식(17-a), 식(17-b)는 아는 값이므로 $F_1(x, t)$ 와 $F_3(x, t)$ 는 적응적으로 조정할 필요는 없다.

식(17-c)에서 α_k 는 미지이므로 α_k 를 스칼라의 가조절 변수 $\delta_k(t)$ 로 바꾸어 쓰면

$$F_{2k}(x, t) = \lambda \delta_k(t) I_n \quad (18)$$

이 성립한다.

식(17-a), 식(17-b), 식(18)를 식(16)에 대입하여 정리하면 다음의 오차방정식을 얻는다.

$$\dot{e} = A_m e + B_m \left(\sum_{k=1}^q \Phi_k g_k \right) \quad (19)$$

단 Φ_k 는 변수오차를 나타내며,

$$\Phi \triangleq (\Phi_1, \Phi_2, \dots, \Phi_q) \\ = (\alpha_1 - \delta_1(t), \alpha_2 - \delta_2(t), \dots, \mathcal{L}_q - \delta_q(t)) \quad (20)$$

로 놓았다.

여기서 Lyapunov의 방법에 의하여 오차방정식(19)가 점근안정이 되도록 Φ_k 를 구하는 과정은 다음과 같다.

Lyapunov 함수를

$$V(t) = e^T(t) P e(t) + \sum_{k=1}^q L_k^{-1} \Phi_k^2 \quad (21)$$

과 같이 정의하면, $P(2n \times 2n)$ 은 Lyapunov 방정식

$$P A_m + A_m^T P = -Q \quad (22)$$

의 유일한 해이며 L_k 는 正의 스칼라로 한다.

$V(t)$ 를 시간미분한 것에 식(19)의 e 를 대입하면

$$\dot{V} = -e^T Q e + 2 \sum_{k=1}^q g_k^T \Phi_k B_m P e \\ + 2 \sum_{k=1}^q \Phi_k L_k^{-1} \dot{\Phi}_k \\ = -e^T Q e + 2 \sum_{k=1}^q \Phi_k (g_k^T B_m^T P e + L_k^{-1} \dot{\Phi}_k) \quad (23)$$

이 되고, 여기서 $\dot{\Phi}_k$ 및 Φ_k 를

$$\Phi_k = -L_k g_k^T B_m^T P e \quad (24-a)$$

$$\Phi_k = -L_k \int_0^t g_k^T B_m^T P e \, d\tau + \Phi_k(0) \quad (24-b)$$

와 같이 놓으면 \dot{V} 는

$$V = -e^T Q e \leq 0 \quad (25)$$

가 되어 점근안정 조건이 성립한다. 여기서 설계 변수 λ 를

$$\lambda > (n+1)m H_{\max}^* \Phi_{\max} \quad (26)$$

로 주면 \ddot{x} 는 有界이고 e 는 연속으로 되며, 여기서 H_{\max}^* 는 $H_k^*(e)$ 의 성분중 최대값이며, Φ_{\max} 는 Φ 의 성분중 최대값이다⁽¹¹⁾.

그리고 하중함수 L_k 와 설계변수 λ 의 관계는 그림 4에서 알 수 있듯이

$$\lambda \geq L_k \quad (27)$$

로 취하여만 제어계가 안정하다.

본 운동상태는 추적초기 위치에서 로봇트 팔은 정지상태인 것에 반하여 컨베이어는 이동중이므로 속도가 일치되기 위하여서는 가속하는 과도상태를 거쳐야 한다. 이 과도특성을 개선하기 위하여 Lyapunov 함수를 이용한다. Lyapunov 함수 $V(t)$ 는 상태공간에서 解의 평형점과의 거리이므로 $-\dot{V}/V$ 를 크게 하면 적응성이 좋아진다⁽¹⁰⁾.

이를 위하여 오차방정식(19)에 보조입력 $W(t)$ 를 넣어서 다음과 같은 새로운 오차방정식을 도출하였다.

오차 e 에 대한 방정식

$$\dot{e} = A_m e + B_m \left(\sum_{k=1}^q \Phi_k g_k - W \right) \quad (28)$$

에서 새로운 Lyapunov 함수로서

$$V(t) = e^T(t) P e(t) + \sum_{k=1}^q L_k^{-1} \Phi_k^2 + W^T L_s^{-1} W \quad (29)$$

와 같이 정하고 위 식을 시간미분하여 식(28)에 대입하면 다음 식이 얻어진다. 여기서 L_s 는 正의 스칼라로 한다.

$$\dot{V} = -e^T(t) Q e(t) + z \left[\sum_{k=1}^q \Phi_k (g_k^T B_m^T P e + L_k^{-1} \dot{\Phi}_k) + W^T (L_s^{-1} \dot{W} - B_m^T P e) \right] \quad (30)$$

여기서

$$\dot{\Phi}_k = -L_k g_k^T B_m^T P e \quad (31 a)$$

$$\dot{W} = L_s (B_m^T P e - L_b W) \quad (31 b)$$

로 취하면

$$\dot{V} = -e^T Q e - 2W^T L_b W \leq 0 \quad (32)$$

로 되어 Φ_k 와 W 를 식(31)과 같이 조정한다면 식(32)로 되어 점근안정으로 된다. 단 L_b 는 정의 스칼라로 한다.

또한 일반적인 설계방식 MRACS와 제안하는 설계방식 PRACS와의 적응성을 비교하기 위하여 Lyapunov 함수 $V(t)$ 와 시간미분 \dot{V} 의 비를 구하면 식(21), (25), 식(28)과 (32)로부터

$$-\frac{\dot{V}_1}{V_1} = \frac{-e^T Q e}{e^T P e + \sum_{k=1}^n L_k \Phi_k^2} = \eta_1 \quad (33)$$

$$-\frac{\dot{V}_2}{V_2} = \frac{e^T Q e + 2W^T L_b W}{e^T P e + \sum_{k=1}^n L_k \Phi_k^2 + W^T L_s W} \quad (34)$$

이며, 여기서 η 는 적응성을 나타낸 것으로 $\eta_2 > \eta_1$ 가 되기 위한 조건은

$$\frac{e^T Q e}{e^T P e} \leq \frac{2W^T L_b W}{W^T L_s W} \quad (35)$$

가 요구된다.

출력 방정식을

$$\begin{aligned} y &= B_m^T P e \\ &= C e \end{aligned} \quad (36 a)$$

와 같이 정의하고 오차방정식 (28)과 결합하면

$$\begin{aligned} \dot{e} &= A_m e + B_m U \\ &= A_m e + B_m \left(\sum_{k=1}^n \Phi_k g_k - W \right) \end{aligned} \quad (36 b)$$

식(36)은 그림 1의 Lure계를 형성한다.

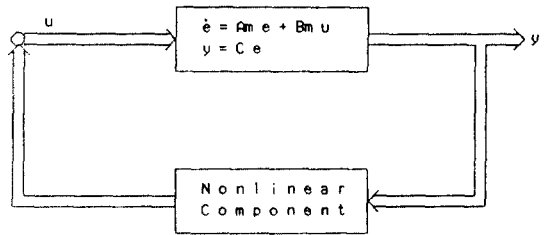


Fig. 1. Lure system of error equation

이 계가 Popov의 초안정정리에 의해 점근초안 정으로 되기 위하여서는 다음의 조건을 만족하지 않으면 안된다⁽¹⁰⁾.

<조건 1> 전달함수 $G(S) = C(SI - A_m)^{-1} B_m$ 은 強正實이다.

<조건 2> 이득 상수 γ_0 가 존재하여

$$\rho(t) \triangleq \int_0^t y^T \left(\sum_{k=1}^n \Phi_k g_k - W \right) d\tau \leq \gamma_0^2, t \geq 0 \quad (37)$$

이다.

<조건 1>은 앞에서 기술한

$$P(A_m + A_m^T P) = -Q \quad (38 a)$$

$$B_m^T P = C \quad (38 b)$$

에 의하여 만족하고, <조건 2>는 식(24)에 의해 만족하고 있다. 그러므로 그림 1의 계통은 점근 초안정이다. 제안하는 강인한 적응제어계의 구성은 그림 2와 같다.

V. 컴퓨터 시뮬레이션

제안한 로봇트 팔의 기능에 관한 시뮬레이션은 그림 3과 같다. 이것은 2링크 로봇트 팔에 대한 것이며 로봇트의 변수 {길이 [m], 질량 [kg]}은 링크 1 [1, 2], 링크 2 [1, 1]이다.

컨베이어의 추적을 위하여 규범모델은 컨베이어로 하고 이 때 컨베이어의 속도는 $0.5 + 0.5t$

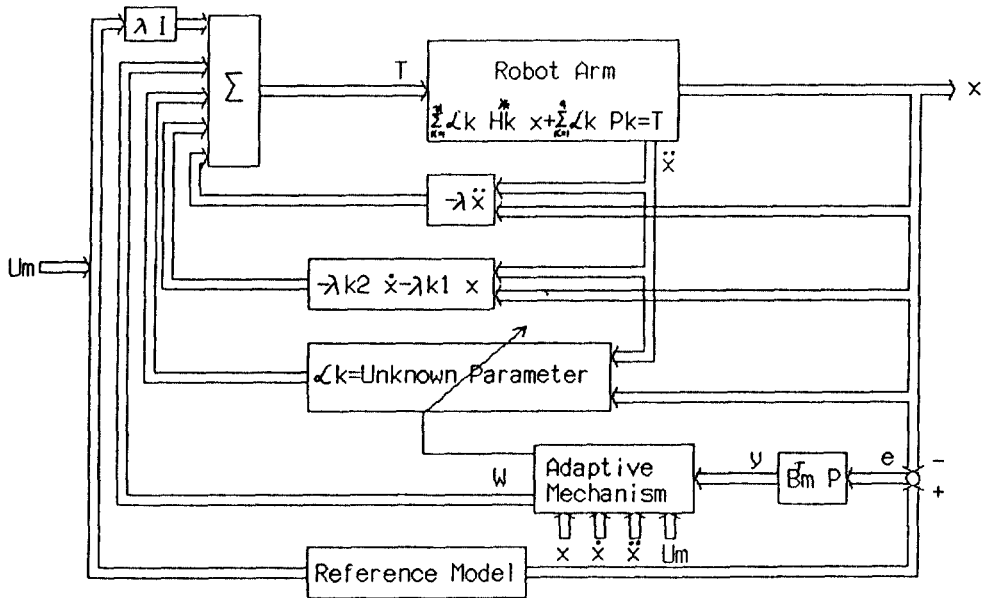


Fig. 2. Robust Adaptive Control System

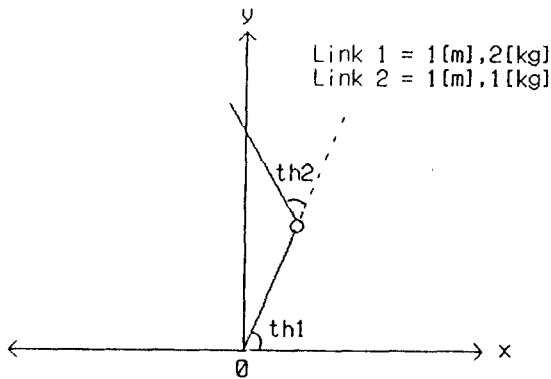


Fig. 3. 2-Link Robot Arm

[m/s]로 주었다. 규범모델 K의 값은 $K_1=K_2=5$ 로 하였고 식(22)의 Q는 $Q=\text{diag}[50, 1]$ 로 택하였다.

또한 컨베이어가 이동한 후 로봇트가 추적하는 것으로 초기의 위치 및 속도의 오차는 각각 $[0.5, 0.0]$ [m], $[0.5, 0.0]$ [m/s]로 주었다.

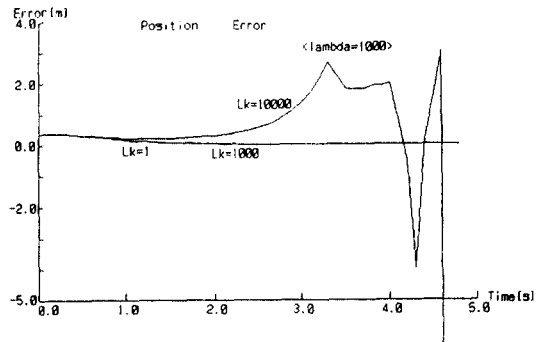


Fig. 4-a Position Error by the Weighting Function(LK)

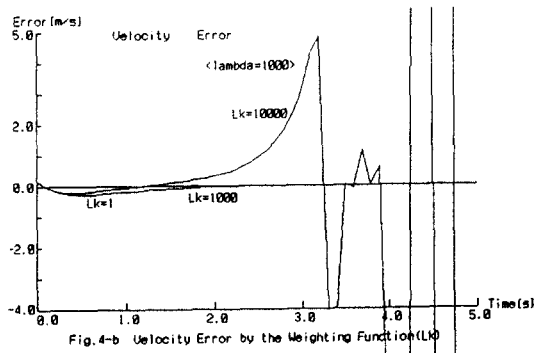


Fig. 4-b Velocity Error by the Weighting Function(LK)

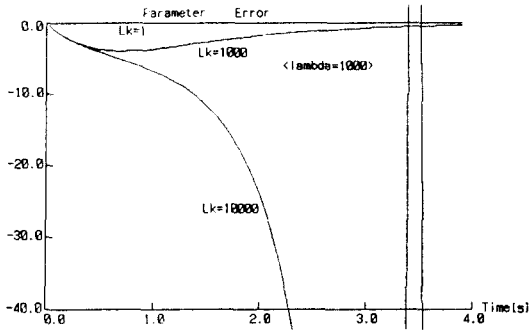


Fig. 4-c Parameter Error by the Weighting Function(LK)

일반적인 MRACS에서 설계변수 λ 을 1000으로 주고 하중함수 L_k 의 값을 각각 1, 1000, 10000로 주었을 때 위치오차, 속도오차, 미지의 매개변수 오차를 그림 4-a, 그림 4-b, 그림 4-c와 같다. 컴퓨터 시뮬레이션의 결과는 L_k 의 값을 λ 의 값보다 작았을 때 오차는 $t=1.7$ sec 부분부터 0가 되어 계통은 안정하나, L_k 의 값이 λ 의 값보다 크면 계통은 불안정하다.

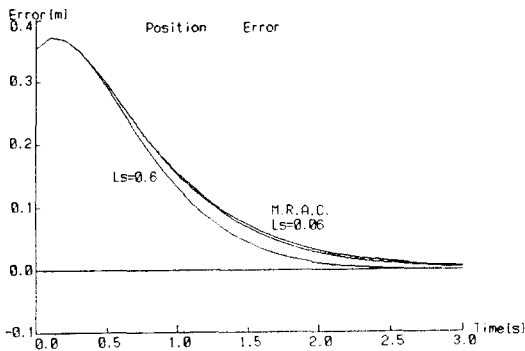


Fig. 5-a Position Error by the Weighting Function(Ls)

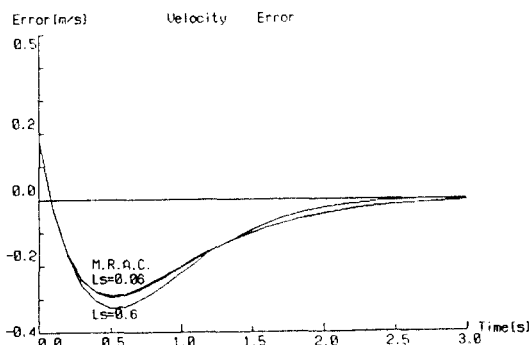


Fig. 5-b Velocity Error by the Weighting Function(Ls)

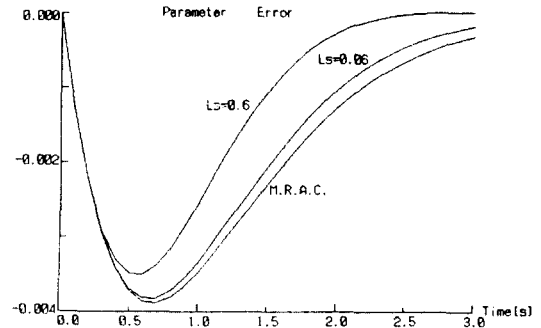


Fig. 5-c Parameter Error by the Weighting Function(Ls)

그림 5 a, 5 b, 5 c는 제안한 설계방식 (PRACS) 와 일반적인 설계 방식(MRACS)에 대해서 L_b 를 1로 정하고 식(35)에 제한되는 하중함수 L_s 의 변화를 변수로한 위치오차, 속도오차, 미지의 매개변수오차를 비교한 것이다. 그림에서 MRACS는 일반적인 설계방식으로 보조 입력이 없으므로 $L_s=0$ 이며 $L_s=0.6, 0.06$ 은 보조 입력을 넣은 PRACS이고, 여기서 식(35)를 만족하는 조건은 $L_s=0.6$ 이다. 이 결과 $L_s=0.6$ 을 가진 PRACS는 MRACS 보다 修正시간이 1 [sec] 이상 빠르다.

그리고 위의 설계조건에서 $\lambda=10000, L_k=1, L_s=6, L_b=0.1$ 로 주었을 때의 결과는 그림 6 a, 6 b, 6 c, 6 d, 6 e이다. 그림 6 a, 6 b에서 M.O는 컨베이어의 이동을 나타내며, 각각 PRACS 와 MRACS에 대하여 위치 및 속도 제어를 비교한 것이다. 여기서도 PRACS가

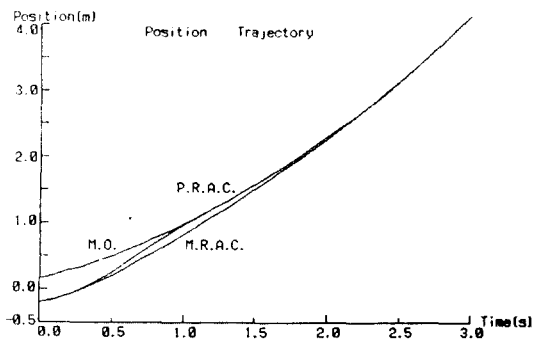


Fig. 6-a Position Trajectory

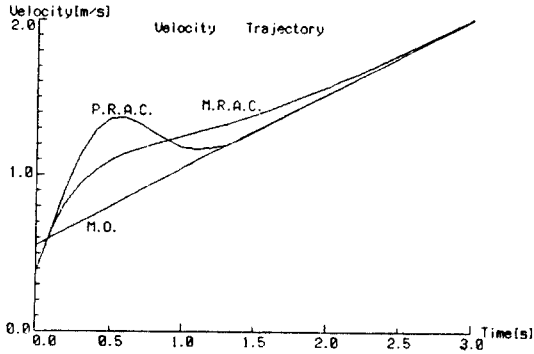


Fig. 6-b Velocity Trajectory

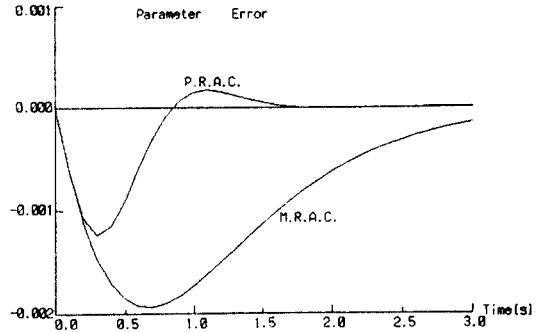


Fig. 6-e Parameter Error

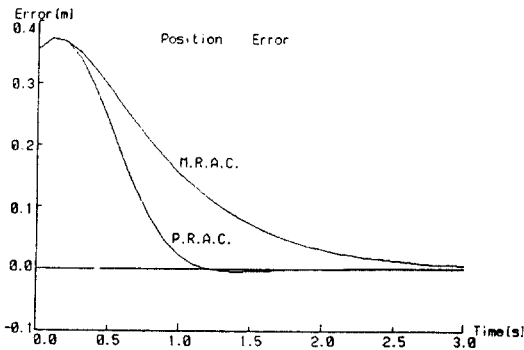


Fig. 6-c Position Error

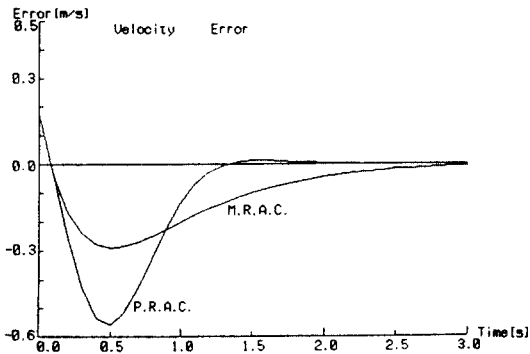


Fig. 6-d Velocity Error

MRACS 에 비하여 修正시간이 1[sec] 이상 빠르며, 그림 6-c, 6-d, 6-e 에서도 같은 결과를 얻었다.

이상과 같이 PRACS는 MRACS 보다 修正시간이 빠르므로 적응성이 좋아져서 과도특성이 개선되는 것을 확인하였다.

VI. 결 론

공장자동화에서 연속작업을 하기 위하여 로봇트 팔이 컨베이어를 추적하는 강인한 적응제어계를 작업좌표계에서 설계하였다. 제안한 설계방식은 모델 규범형 설계방식과 Popov의 초안정정리를 기본으로 하였고 보조입력을 사용하여 Lyapunov 함수 $V(t)$ 와 시간미분 \dot{V} 와의 비, 즉 $-\dot{V}/V$ 의 크기를 크게 하여 일반적인 모델규범형 설계방식보다 과도특성을 개선하였다.

또한 일반적인 모델규범형 설계방식에서 설계변수 λ 와 하중함수 L_k 는 L_k 의 값이 λ 의 값보다 작은 경우는 계통이 안정하나 L_k 의 값이 λ 의 값보다 큰 경우는 계통이 불연속적으로 불안정하게 되었다.

컴퓨터 시뮬레이션에 의하여 이상과 같은 특성을 확인하였다.

參 考 文 獻

1. Shin Ichi Miyamoto, Seichi Shin, Shigenori Okubo and Toshiyuki Kitamori, "Continuous time model reference adaptive control based on weighting functions", IEEE, Trans., Vol. AC-32, No. 5, pp. 442~447, May 1987.
2. A. Feuer, "A parametrization for model reference adaptive pole placement", IEEE, Trans., Vol. AC 31, No. 2, pp. 782~785, February 1986.
3. G. Kreisselmeier and B.O. Anderson, "Robust model reference adaptive control", IEEE, Trans., Vol. AC 31, No. 2, pp. 127~133, February 1986.
4. 宮里義彦, 大島康次郎, "Robust manipulator의 非線形適應制御", 計測自動制御學會論文集, 24 1, pp. 63~68, January 1988.
5. 小笠一弘, 吉田勝久, 橫山龍昭, "Robot의 假想內部 model 追從制御系", 計測自動制御學會論文集, 24 1, pp. 55~62, January 1988.
6. Sahjendra N. Singh, "Adaptive model following control of nonlinear robotic systems", IEEE, Trans., Vol. AC-30, No. 11, pp. 1099~1100, November 1985.
7. C. S. G. Lee and M. J. Chung, "An adaptive control strategy for mechanical manipulators", IEEE, Trans., Vol. AC 29, No. 9, pp. 837~840, September 1984.
8. S. Nicisia and P. Tomei, "Model reference adaptive control algorithms for industrial robots", Automatica, Vol. 20 5, pp. 635~644, May 1984.
9. Anni J. Koivo and Ten Huel Guo, "Adaptive linear controller for robotic manipulators", IEEE, Trans., Vol. AC 28, No. 2, pp. 162~171, February 1983.
10. 半井一正, 池田雅夫, "非線形制御 system의 解釋", Ohm 社, pp. 81~117, 1986.
11. 美多勉, 天伯賀公三, "Robot 制御工学入門", Corona 社, pp. 215~226, 1989.
12. 金井喜美雄, "Robust 適應制御 入門", Ohm 社, pp. 90~115, pp. 236~239, 1989.



嚴 基 煥 (Ki Hwan EOM) 正會員
 1949年 10月 8日生
 1972年 2月 25日 : 東國大學校 電子工學
 科 卒業 (工學士)
 1975年 8月 30日 : 仁荷大學校 大學院 電
 氣工學科 卒業 (工學碩士)
 1986年 2月 25日 : 東國大學校 大學院 電
 子工學科 卒業 (工學博士)
 1978年 2月 - 現在 : 柳韓工業專門大學 電
 氣科 教授



孫 東 壽 (Dong Seol SON) 正會員
 1952年 8月 18日生
 1978年 2月 25日 : 東國大學校 電子工學
 科 卒業 (工學士)
 1981年 2月 25日 : 東國大學校 大學院 電
 子工學科 卒業 (工學碩士)
 1990年 3月 - 現在 : 東國大學校 大學院
 電子工學科 博士過程
 1989年 3月 - 現在 : 柳韓工業專門大學 電
 子工學科 兼任講師

김 주 흥 (Ju Hong KIM) 正회원
 동국대학교 전자공학부 교수