

양팔 로봇의 제어에 관한 연구

김 중현, 김 중삼, 박 만식, 이 석규, 배 진호

A study on Control of a Dual-arm Robot

Jong-hyun Kim, Jong-sam Kim, Man-sik Park, Suck-gyu Lee, Jin-ho Bae

Department of Electrical Engineering, Yeungnam University

Abstract

Coordinated motion control of two arms of a dual-arm robot has been studied by many researchers, because of its potential application in assembly as well as the handling of large and heavy objects beyond the capacity of single arm. This paper derives dynamic equation of a dual-arm robot, and describes some constrains to pick up a simply shaped object at prespecified position on it. This paper concludes with describing both PD and self-tuning control algorithm for the above task.

1. 서론

일반적으로, 양팔 로봇은 큰 중량의 대상물을 잡고, 일정한 방향으로 움직이며 비교적 쉽게 무게중심을 잡을 수 있는 장점이 있다. 또한 하나의 공동제어기를 사용하여 양팔을 제어할 수 있으므로, 그 작업시간이 단축되고 제어가 비교적 쉽게 된다. 최근에는 이러한 복수의 팔을 가진 robot에 관한 연구[2], [3], [6], [7], 혹은 복수의 robot가 서로 협조하면서 공동의 작업을 수행하는데 관한 연구[1], [4], [5]가 차츰 활기를 띠고 있다. Kosuge[1], [4] 등은 virtual internal model을 사용하여 복수의 팔을 가진 robot를 제어하는 새로운 방법을 제시하였으며, Wang[2]은 우주 공간에서 양팔을 가진 robot의 간단한 control strategy를 제안하였다. 또한 Ishida[3]는 두 팔을 이용한 force control에 대해서, Luh[2] 등은 2대의 robot의 효율적인 제어법을 제시하였다.

본 논문에서는, 이러한 우주공간에서 작업하는 양팔 로봇가 단순한 형태의 대상물을 잡을 때에 고려되어야 할 기본적인 조건들을 제시하며, 로봇 시스템 control에 있어서 PD control 및 몇몇 parameter에 대한 정보가 부족한 경우를 가정한 self-tuning controller를 구성하였다.

2. 문제의 설정

그림1에 나타난 것처럼 우주공간에서 단순작업을 하는 양팔 로봇가 단순한 형태을 한 물체를 주어진 위치에 end-effector를 위치 시키고자 한다.

실제로는 자유도가 양팔에 대해서 4개, main body의 rotation에 관해 3개, translational movement에 대해서 3개 등 모두 10 자유도가 되나, 여기서는 전체적인 흐름을 해치지 않는 범위 내에서 수식적으로 간단하게 하기 위해 양팔에 대한 4 자유도, main body의 rotation에 대해 1 자유도, 그리고 평면상을 움직인다고 가정을 하여 2 자유도 등 모두 7 자유도에 대하여 생각을 한다.

일반적으로 robot의 움직임에 대하여 연구할 때, 제일 먼저 고려하여야 할 사항은 물체와 로봇과의 충돌회피이나, 양팔 로봇의 경우에는 양팔 끼리의 충돌회피 및 양팔과 main body와의 경우도 고려하여야 하며, 이것을 수학적으로 아래와 같이 나타낼 수 있다.

(i) 작업 대상물과 로봇과의 충돌회피 조건

$$\partial \Sigma_R(t) \cap \partial \Sigma_O(t) = \phi$$

여기서 $\partial \Sigma_R(t)$ 및 $\partial \Sigma_O(t)$ 는 시간 t에서의 robot 및 작업 대상물의 경계면을 나타낸다.

(ii) 각 link 끼리의 충돌회피 조건

$$\bigcap_i \partial \Sigma_{ii}(t) = \phi \quad (i: \text{각 link})$$

여기서 $\partial \Sigma_{ii}(t)$ 는 시간 t에서의 link i의 경계면을 나타낸다.

(iii) 각 link와 main body와의 충돌회피 조건

$$\partial \Sigma_{ii}(t) \cap \partial \Sigma_B(t) = \phi$$

여기서 $\partial \Sigma_B(t)$ 는 시간 t에서의 main body의 경계면이다. 또한 실제 작업시, 대상물을 잡을 때는 oscillation이 없이 작업을 할 수 없도록하여야 한다.

주어진 작업은, 임의의 초기상태의 로봇트가 지정된 막대 모양의 물체와 평행하게 회전한 다음에 막대의 길이 방향으로 움직여 막대의 중심과 main body의 중심을 일치 시킨 후, 다시 양 팔의 end-effector가 물체에 표시된 지점에 일치하도록 위치시키는 것이다. 그 다음, 로봇트를 막대의 중심 방향으로 서서히 옮기는 것이다. 이러한 일련의 동작은 그 효율적인 면에서 좋지 못하나, 연구 목적상 간단한 경로를 택하였다. 실제 작업 특히 공장에서의 작업은 그 농밀을 고려하여 경로 계획이 되어져야 한다.

3. 로봇트의 동적 방정식

일반적으로, 로봇트의 운동방정식을 구하는 데는 Newton-Euler formulation 및 Lagrangian formulation이 있다. Lagrangian formulation은 generalized coordinates를 사용하여 주어진 system의 동적인 움직임을 일과 에너지의 항으로 나타내어 로봇트의 dynamic equation을 유도한다. 이 방법은 Newton의 운동 제2법칙을 이용한 Newton-Euler formulation에 비해 비교적 간단히 체계적으로 동적 방정식을 유도할 수가 있으며, 그 결과식이 일반적으로 간단하고, 관절토크와 변위의 closed form으로 나타나므로, 여기서는 이 방법을 이용한다.

Lagrangian T는 주어진 system의 운동에너지 K와 위치에너지 P와의 차이로서, 식(1)과 같이 나타내어 진다.

$$T = K - P \quad (1)$$

그러나 여기서는 중력이 없는 우주공간에서의 robot의 움직임에 관한 것이므로, P = 0 라 가정한다.

따라서

$$T_i = \frac{1}{2} I_i \dot{\theta}_i^2 + \frac{1}{2} m_i v_i^2 \quad (i=1,2,\dots,7) \quad (2)$$

와 같이 표시되며, 여기서 I_i , m_i , v_i , θ_i 는 link i에 관한 관성 모멘트, 질량, 무게중심의 속도 및 각속도를 각각 나타낸다.

따라서

$$\begin{aligned} T_0 &= \frac{1}{2} I_0 \dot{\theta}_0^2 + \frac{1}{2} m_0 v_0^2 \\ &= \frac{1}{2} I_0 \dot{\theta}_0^2 + \frac{1}{2} m_0 [\{ x - (y - y_0, \cos) \dot{\theta}_0 \}^2 \\ &\quad + \{ y + (x + x_0, \cos) \dot{\theta}_0 \}^2] \end{aligned}$$

$$\begin{aligned} T_1 &= \frac{1}{2} I_1 (\dot{\theta}_0 + \dot{\theta}_1)^2 + \frac{1}{2} m_1 v_1^2 \\ &= \frac{1}{2} I_1 (\dot{\theta}_0 + \dot{\theta}_1)^2 + \frac{1}{2} m_1 [\{ x - \theta_1 l_{c1} \sin \theta_1 \\ &\quad - (y + l_{c1} \sin \theta_1) \dot{\theta}_0 \}^2 + \{ (x + l_{c1} \cos \theta_1) \dot{\theta}_0 + y \\ &\quad + \theta_1 l_{c1} \cos \theta_1 \}^2] \end{aligned}$$

$$\begin{aligned} T_2 &= \frac{1}{2} I_2 (\dot{\theta}_0 + \dot{\theta}_1 + \dot{\theta}_2)^2 + \frac{1}{2} m_2 v_2^2 \\ &= \frac{1}{2} I_2 (\dot{\theta}_0 + \dot{\theta}_1 + \dot{\theta}_2)^2 \\ &\quad + \frac{1}{2} m_2 [\{ x - \theta_1 l_1 \sin \theta_1 \\ &\quad - (\dot{\theta}_1 + \dot{\theta}_2) l_{c2} \sin(\theta_1 + \theta_2) - (y + l_1 \sin \theta_1 \\ &\quad + l_{c2} \sin(\theta_1 + \theta_2)) \dot{\theta}_0 \}^2 + \{ (x + l_1 \cos \theta_1 \\ &\quad + l_{c2} \cos(\theta_1 + \theta_2)) \dot{\theta}_0 + y + \theta_1 l_1 \cos \theta_1 \\ &\quad + (\dot{\theta}_1 + \dot{\theta}_2) l_{c2} \cos(\theta_1 + \theta_2) \}^2] \end{aligned}$$

$$\begin{aligned} T_1' &= \frac{1}{2} I_1' (\dot{\theta}_0 + \dot{\theta}_1')^2 + \frac{1}{2} m_1' v_1'^2 \\ &= \frac{1}{2} I_1' (\dot{\theta}_0 + \dot{\theta}_1')^2 \\ &\quad + \frac{1}{2} m_1' [\{ x - \theta_1' l_{c1}' \sin \theta_1' \\ &\quad - (y + l_{c1}' \sin \theta_1') \dot{\theta}_0 \}^2 + \{ (x + \omega \\ &\quad + l_{c1}' \cos \theta_1') \dot{\theta}_0 + y + \theta_1' l_{c1}' \cos \theta_1' \}^2] \end{aligned}$$

$$\begin{aligned} T_2' &= \frac{1}{2} I_2' (\dot{\theta}_0 + \dot{\theta}_1' + \dot{\theta}_2')^2 + \frac{1}{2} m_2' v_2'^2 \\ &= \frac{1}{2} I_2' (\dot{\theta}_0 + \dot{\theta}_1' + \dot{\theta}_2')^2 \\ &\quad + \frac{1}{2} m_2' [\{ x - \theta_1' l_1' \sin \theta_1' \\ &\quad - (\dot{\theta}_1' + \dot{\theta}_2') l_{c2}' \sin(\theta_1' + \theta_2') \\ &\quad - (y + l_1' \sin \theta_1' + l_{c2}' \sin(\theta_1' + \theta_2')) \dot{\theta}_0 \}^2 \\ &\quad + \{ (x + \omega + l_1' \cos \theta_1' \\ &\quad + l_{c2}' \cos(\theta_1' + \theta_2')) \dot{\theta}_0 + y + \theta_1' l_1' \cos \theta_1' \\ &\quad + (\dot{\theta}_1' + \dot{\theta}_2') l_{c2}' \cos(\theta_1' + \theta_2') \}^2] \end{aligned}$$

여기서 T = T₀ + T₁ + T₂ + T_{1'} + T_{2'} 이라면, dynamic equation은 식(3)과 같이 나타내어 진다.

$$F_i = \frac{d}{dt} \frac{\partial T}{\partial \dot{q}_i} - \frac{\partial T}{\partial q_i} \quad (3)$$

여기서 q_i 는 generalized coordinate를 나타내며, \dot{q}_i 는 그 속도, F_i 는 generalized force 혹은 Torque를 나타낸다. 식(3)을 정리하면 식(4)를 얻을 수가 있다.

$$H(q) \dot{q} + V(q, \dot{q}) = u \quad (4)$$

$$q = (\theta_1, \theta_2, \theta_1', \theta_2', \theta_0, x_0, y_0)^T$$

$$u = (u_1, \dots, u_7)^T \triangleq (\tau_1, \tau_2, \tau_1', \tau_2', \tau_c, f_{cx}, f_{cy})^T$$

여기서 H(q)는 관성항이며, V(q, \dot{q})는 원심력과 Coriolis력을 나타내는 항이다.

$$V = (V_1, \dots, V_7)^T \quad (5)$$

$$H = \begin{bmatrix} h_{11} & h_{12} & 0 & 0 & h_{15} & h_{16} & h_{17} \\ h_{12} & h_{22} & 0 & 0 & h_{25} & h_{26} & h_{27} \\ 0 & 0 & h_{33} & h_{34} & h_{35} & h_{36} & h_{37} \\ 0 & 0 & h_{43} & h_{44} & h_{45} & h_{46} & h_{47} \\ h_{51} & h_{52} & h_{53} & h_{54} & h_{55} & h_{56} & h_{57} \\ h_{16} & h_{26} & h_{36} & h_{46} & h_{56} & h_{66} & 0 \\ h_{17} & h_{27} & h_{37} & h_{47} & h_{57} & 0 & h_{77} \end{bmatrix} \quad (6)$$

위의 로봇의 동적방정식의 해를 구하기 위해서, 이것을 상태공간법으로 표시하면, $H(q)$ 는 행렬의 성질상 특이 행렬이 아니므로 역변환이 존재하므로 아래와 같이 나타낼 수 있다.

$$\ddot{q} = H^{-1}(q)(u - V(q, \dot{q})) \quad \text{이므로}$$

x_1 및 x_2 는 각각 아래와 같이 정의한다.

$$x_1 = q, \quad x_2 = \dot{q}$$

여기서 $q = (\theta_1, \theta_2, \theta_1', \theta_2', \theta_0, x, y)^T$ 이다.

따라서

$$x_1 = x_2$$

$$x_2 = H^{-1}(x_1)(u - V(x_1, x_2))$$

로 나타내어진다. (x_1, x_2 는 7×1 vector)

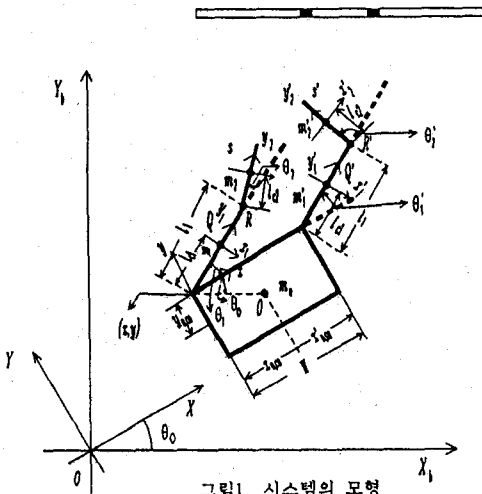


그림1. 시스템의 모형

4. Control algorithm

4.1 PD control

Robot control을 위한 가장 단순하면서도 효과적인 방법은 PD control 이다. 이때 P는 개루우프 정상상태 이득이며, D는 응답의 오버 슈루우트를 감소시키는 예측 동작을 행한다.

$$q(t) = (\theta_1, \theta_2, \theta_1', \theta_2', \theta_0(t), x, y)^T \triangleq \dot{q}(t)$$

$$\dot{q}(t) = (0, 0, 0, 0, \dot{\theta}_0(t), 0, 0)^T \triangleq \ddot{q}(t) \quad (7)$$

$$\ddot{q}(t) = (0, 0, 0, 0, \ddot{\theta}_0(t), 0, 0)^T \triangleq \ddot{\dot{q}}(t)$$

여기서 $x, y, \theta_i, (i=1, 2, 1', 2')$ 는 지정된 상수이다.

따라서 식(4)는 다음과 같이 줄일 수 있다.

$$h_{55}(q) \ddot{\theta}_0 = -V_5(q, \dot{q}) + \tau_c, \quad (8)$$

$$h_{is}(q) \ddot{\theta}_0 = -V_i(q, \dot{q}) + u_i, \quad i \neq 5. \quad (8)$$

여기서 막대와 robot가 평행하게 되는 각도를 θ_0^d 라 하고,

$\Delta \theta_0 \triangleq \theta_0^d - \theta_0$ 하면 식(8)은 다음과 같이 표현할 수 있다.

$$h_{55}(q) \Delta \ddot{\theta}_0 = V_5(q, \dot{q}) + \tau_c \quad (9)$$

식(9)에서

$$\tau_c = V_5(q, \dot{q}) + h_{55}(q) (K_p \Delta \theta_0 + K_v \Delta \dot{\theta}_0), \quad (10)$$

여기서 K_p 와 K_v 는 feedback gain의 상수이다.

따라서 robot가 막대와 평행하게 움직이는 동안 모든 조인트 각각과 base position이 고정되게 하기위한 관절 토크 $\tau_i (i=1, 2, 1', 2')$ 와 base control force f_c 는 다음과 같이 주어진다.

$$u_i = V_i(q, \dot{q}) + h_{is}(q)(K_p \Delta \theta_0 + K_v \Delta \dot{\theta}_0), \quad i \neq 5. \quad (11)$$

알고리즘을 요약하면 다음과 같다.

Step1: 초기치 $q_i(\theta), q_i(0), \dot{q}_i(0)$ 을 선택하고, $i=1, \dots, 7$

파라메타 K_p, K_v 를 임의의 값으로 정한다.

Step2: 식(10)과 식(11)을 이용하여 입력 U_i 를 계산한다.

Step3: 계산된 입력 U_i 를 system에 적용하여 측정치를 얻고, desired 한 값과 비교하여 $\Delta \theta_1$ 를 구한 다음 샘플링 구간을 다음 구간으로 바꾼다.

Step4: Step2를 최종시간까지 반복한다.

4.2 Self-tuning controller

System의 parameter가 미지이거나, 갑작스런 부하의 변동과 같은 제어환경이 변화하는 경우에는 PD 제어기와 같은 고정 이득으로 설계된 선형제어기로는 만족할 만한 성능을 얻을 수 없다. 이와 같은 제어기의 parameter들을 on-line으로 자동 조절하는 self-tuning controller는 매우 효과적이다. 일반적으로 self-tuning control에서의 secondary controller는 아래와 같다.

$$u_i(k) = \frac{b_0^i}{(b_0^i)^2 + \epsilon} [y_i^d(k+1) - z^{-1} A_{11}(z^{-1}) y_i(k) - a_0^i - B_{11}(z^{-1}) u_i(k)] \quad (12)$$

여기에서

$$A_{11}(z^{-1}) = a_{11}^i z^{-1} + \dots + a_{n_1}^i z^{-1}$$

$$B_{11}(z^{-1}) = b_{11}^i z^{-1} + \dots + b_{n_1}^i z^{-1}$$

또한 최적 평가 파라메타 $\alpha_1(k)$ 는 다음과 같다.

$$\alpha_i = [a_1^i \dots a_2^i ; b_0^i \dots b_n^i ; a_0^i]$$

α_i 를 on-line 연산을 위한 결과식은 다음과 같다.

$$\alpha_i(k) = \alpha_i(k-1) + P_i(k)\phi_i(k-1) [y_i(k) - \phi_i^T(k-1)\alpha_i(k-1)] \quad (13)$$

$$P_i(k) = \frac{1}{\gamma_i} \left[P_i(k-1) - \frac{P_i(k-1)\phi_i(k-1)\phi_i^T(k-1)P_i(k-1)}{\gamma_i + \phi_i^T(k-1)P_i(k-1)\phi_i(k-1)} \right] \quad (14)$$

여기서

$$\phi_i^T(k-1) = [y_i(k-1) \dots y_i(k-2) ; u_i(k-1) \dots u_i(k-n-1) ; 1]$$

식(7)의 coliolis 항이 known하다고 가정하면 primary controller와 secondary controller는

$$U_r^d(t) = V(q^d, \dot{q}^d): \text{form primary controller} \quad (15)$$

$$U(t) = H(q) \ddot{q} \quad (16)$$

라고 둘 수 있다.

따라서 system 입력 $U(t)$ 는 다음과 같다.

$$U(t) = U_r^d(t) + U(t) \quad (17)$$

전체 self-tuning controller의 알고리즘을 요약하면 다음과 같다.

- Step1: 초기값 $\alpha_i(0)$, $P_i(0)$, $U_i(0)$, $U_i(k-n-1)$ 를 계산하고 ϵ_i , γ_i 를 임의의 값으로 선택한다.
- Step2: 식(13), 식(14)를 이용하여 파라메타 $\alpha_i(k)$ 를 계산하고, 식(12)과 식(17)을 이용하여 $U_i(t)$ 를 구한다.
- Step3: 계산된 $U_i(t)$ 를 system에 입력하여 inverse dynamic 을 계산하여 K 를 $K+1$ 로 대체한다.
- Step4: Step2를 최종시간까지 반복한다.

4.3 시뮬레이션

시뮬레이션은 robot가 초기위치에서 막대를 잡는 위치까지 이동하는 상황을 4단계로 고려하였다. 표 1의 초기 토오크 및 초기위치와 최종위치는 임의의 수치값을 대입하였으며, 각 수치는 각단계의 상황을 만족한다고 가정한다.

- 단계 1: robot의 초기 위치에서 막대와 평행한 위치까지 base축을 회전운동한다.
- 단계 2: 막대의 무게중심과 base중심을 일치시키기 위해 X축으로 병진운동한다.
- 단계 3: arm이 막대를 잡을 수 있는 위치까지 Y축으로 병진운동한다.
- 단계 4: arm이 막대를 잡을 수 있는 위치로 관절각을 조절한다.

표 1 초기조건

	초기위치	최종위치	초기토오크	시간
θ_1	60 [deg]	40 [deg]	0.03	0 - 10 [sec]
θ_2	30 "	50 "	0.03	10 - 20 "
θ_1'	50 "	70 "	0.03	20 - 30 "
θ_2'	20 "	50 "	0.03	30 - 40 "
θ_0	10 "	0 "	0.5	40 - 50 "
x_0	30 [cm]	60 [cm]	0.03	50 - 60 "
y_0	40 "	80 "	0.03	60 - 70 "

시뮬레이션에서 사용된 파라메타의 값들은 다음과 같다.
 $l_1 = l_1' = 3, l_2 = l_2' = 2, l_{c1} = l_{c1}' = 1.5, l_{c2} = l_{c2}' = 1.0, I_0 = 1.0, I_1 = I_2 = I_1' = I_2' = 0.1, m_0 = 20.0, m_1 = m_1' = 1.5, m_2 = m_2' = 1.0, x_{c0} = 3.5, y_{c0} = 1.5, w = 7.0$

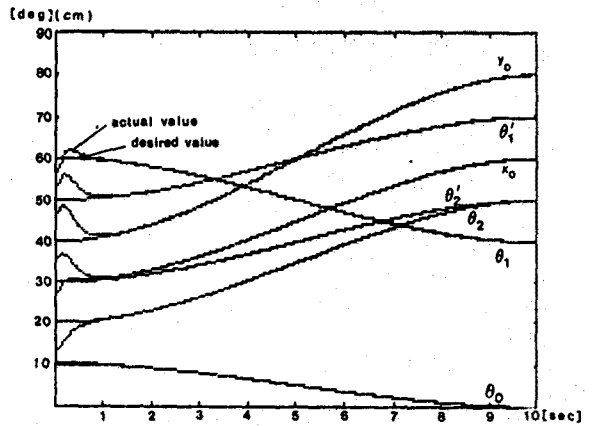


그림2. 각 coordinate의 disired value와 actual value

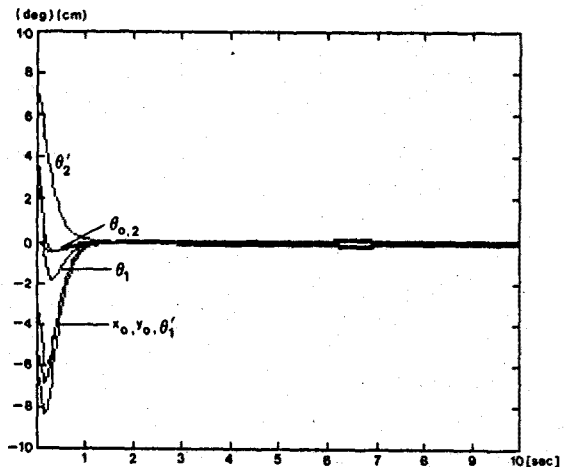


그림3. 각 coordinate의 오차

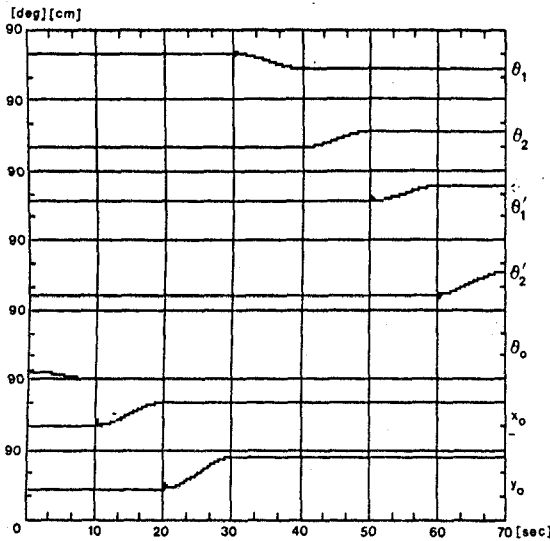


그림4. 각 coordinate의 time chart

그림2는 각 단계에서의 $\theta_1, \theta_2, \theta_1', \theta_2', \theta_0, x_0, y_0$ 를 PD control을 사용해 나타내었다. 그림3는 각 변위에서의 desired 한 값과 실제값의 차를 나타내었다. 그림4는 단계 4 까지의 각 변위를 0-70 초 까지의 전체 변화를 나타내었다.

5. 결론

본 연구에서는 우주공간에서 작업하는 단순한 형태를 가진 양팔 로봇의 작업시 고려 되어야 될 조건 및 controller 구성에 대하여 서술하였다. 이러한 것은 단지 중력항만을 더 고려해 줌으로써 공장에서도 실제로 사용될 수 있으리라 생각된다. 이 경우 작업의 능률 향상을 위한 optimal controller 등과 같은 작업에 따라 적절한 controller algorithm을 개발하는 것이 중요하다.

6. 참고 문헌

[1] K. Kosuge, J. Ishikawa, K. Furuta and M. Sakai, "Control of Single-master Multi-slave Manipulator System Using VIM", Proc. IEEE Intl. Conf. on Robotics and Automation, pp.1172-1177, 1990.

[2] J.M. Tao, J.Y.S. Luh and Y.F. Zheng, "Compliant Coordination Control of Two Moving Industrial Robots", IEEE Trans. of Robotics and Automation, Vol.6, No.3, pp.322-330, 1990.

[3] A.M.Dudar and A.H.Eltimsahy, "A Near-minimum Time Controller for Two Coordinating Robots Grasping an Object", Proc. IEEE Intl. Conf. on Robotics and Automation, pp.1184-1189, 1990.

[4] K. Kosuge, M. Koga, K. Furuta and N. Nosaki, "Coordinated Motion Control of Robot Arm Based on Virtual Internal Model", Proc. IEEE Intl. Conf. on Robotics and Automation, pp.1097-1102, 1989.

[5] T.J.Tarn, A.K.Bejczy and X.Yun, "Design of Dynamic Control of Two Cooperating Robot Arms: Closed Chain Formulation" Proc. IEEE Intl. Conf. on Robotics and Automation, pp.7-13, 1987.

[6] M.A.Unseren and A.J.Koivo, "Reduced Order Model and Decoupled Control Architecture for Two Manipulators Holding an Object", Proc. IEEE Intl. Conf. on Robotics and Automation, pp.1240-1245, 1989.

[7] Y.Nakamura, K.Nagai and T.Yoshigawa, "Dynamics and Stability in Coordination of Multiple Robotic Mechanisms", Intl. Journal of Robotics Research, Vol. 8, No.2, pp.44-61, 1989.

[8] P.K.C.Wang, "Control Strategy for a Dual-arm Maneuverable Space Robot", Proc. of the Workshop on Space Telerobotics, JPL Pub. 87-13, Vol.2, pp.257-266, 1987.

[9] T.Ishida, "Force Control in Coordination of Tow Arms", proc. of 5th Intl. Joint Conf. on Artificial Intelligence, pp.717-722, 1977.