

박 현 우 , 배 준 경 , 박 중 국

경희 대학교 전자 공학과

CARTESIAN MOTION CONTROL OF MANY DEGREE OF FREEDOM MANIPULATOR

Hyun-Woo Park , Jun-Kyung Bae , Chong-Kuk Park

Kyung Hee University

ABSTRACT

The basic concept of this control scheme is that of an ideal massless manipulator carrying a load. The purpose of this paper is to converge the observed forces and moments at the mass center to the desired applied forces and moments.

And the corresponding control algorithm is proposed.

I. 서 론

Cartesian 영역에서 로봇 매니플레이터의 운동을 제어하기 위해서는 이를 조인트 영역에서의 운동으로 바꾸어 줘야 한다.

오늘날 Cartesian 운동을 조인트 운동으로 변환시켜 주는 방법으로는 역 Jacobian 행렬을 이용하는 방법이 사용되고 있다. 본 논문에서는 부하의 질량 중심에 대한 원하는 궤적 값이 주어지며, 질량 중심의 Cartesian 좌표계에서 발생하는 위치와 속도 오차를 계산함으로써 질량 중심 주위에서의 간단한 위치 제어가 가능해진다. 이러한 Cartesian 오차를 보상하기 위해 질량 중심에 인가되어야 하는 원하는 힘과 모멘트는 이러한 제어기에 의해 계산된다.

매니플레이터는 질량을 갖고있지 않기 때문에 일고자 하는 힘과 모멘트는 $6 \times n$ Jacobian 행렬의 전치 형태(Transpose)를 이용하여, 일고자 하는 조인트 모오드로 변환시킬 수 있다.

여기서 n 은 매니플레이터의 조인트 수를 나타낸다

II. 질량 중심의 위치 제어

위치와 방향 오차를 수정하기 위해, 질량 중심에 인가되는 일고자하는 힘과 모멘트를 계산해야 하기 때문에 질량 중심의 위치 제어가 필요하다. 우리는 먼저 control point 의 위치와 속도에 대한 오차를 구한 후 이러한 오차를 질량 중심의 위치와 속도에 대한 오차들로 변환시켜야 하며, 이에 대한 제어기법은 다음과 같다.

1. 질량 중심의 위치 오차를 계산한다.

실제 Revolute 각과 Prismatic 간격은 측정 가능하며 실제 변환 T_{na} 는 링크 변환들을 모두 곱함으로써 계산된다. 따라서 위치와 방향오차 T_n 은 다음과 같이 얻어질 수 있다. [3]

$$\delta T_n = T_{nd}^{-1} \times T_{na} \quad \text{----(1)}$$

여기서 T_{nd} : 일고자 하는 조인트 변환
 T_{na} : 실제 조인트 변환

$$\delta T_n = \begin{bmatrix} 1 & -\delta Z_n & \delta Y_n & \Delta X_n \\ \delta Z_n & 1 & -\delta X_n & \Delta Y_n \\ -\delta Y_n & \delta X_n & 1 & \Delta Z_n \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----(2)}$$

이며, $\Delta X_n, \Delta Y_n, \Delta Z_n, \delta X_n, \delta Y_n, \delta Z_n$ 은 T_{nd} 의 기본축에 대한 세 변위와 세 회전을 나타낸다. 이때 이러한 여섯개의 원소는 다음과 같이 간단한 형태로 나타낼 수 있다.

$$\begin{bmatrix} \Delta X_n \\ \Delta Y_n \\ \Delta Z_n \\ \delta X_n \\ \delta Y_n \\ \delta Z_n \end{bmatrix} = \begin{bmatrix} (P_a - P_d) \cdot R_d \\ (P_a - P_d) \cdot O_d \\ (P_a - P_d) \cdot \Delta d \\ O_a \cdot \Delta d \\ O_a \cdot R_d \\ R_a \cdot O_d \end{bmatrix} \quad \text{----(3)}$$

이러한 오차들을 질량 중심에서의 오차로 변환하면

$$\begin{bmatrix} \Delta X_m \\ \Delta Y_m \\ \Delta Z_m \\ \delta X_m \\ \delta Y_m \\ \delta Z_m \end{bmatrix} = \begin{bmatrix} n \cdot d + (P \times n) \cdot \delta \\ o \cdot d + (P \times o) \cdot \delta \\ a \cdot d + (P \times a) \cdot \delta \\ n \cdot \delta \\ o \cdot \delta \\ a \cdot \delta \end{bmatrix} \quad \text{----(4)}$$

여기서, $\Delta X_m, \Delta Y_m, \Delta Z_m, \delta X_m, \delta Y_m, \delta Z_m$ 은 질량 중심의 위치 오차이며, n, o, a, p 는 질량 중심의 기본 속과 end-point 의 좌표계 사이의 등차 변환 행렬의 세 원소 열이다. 그리고 d 는 식(3) 의 $[\Delta X_n \ \Delta Y_n \ \Delta Z_n]^T$, δ 는 식(3) 의 $[\delta X_n \ \delta Y_n \ \delta Z_n]^T$ 이다.

2. 질량 중심의 속도 오차를 계산한다.

앞의 위치 오차를 계산한 행렬에서 속도 오차를 유도할 수 있다. 시간 t 에서 질량 중심의 속도 오차는 다음과 같이 나타낸다.

$$\dot{X}_{me}(t) = \dot{X}_{ma}(t) - \dot{X}_{md}(t) \quad \text{----(5)}$$

여기서,

\dot{X}_{ma} : 실제 Cartesian 속도

\dot{X}_{md} : 링크자 하는 Cartesian 속도

이것은 각각

$$\dot{X}_{ma} = \begin{bmatrix} \dot{X}_{ma} \\ \dot{Y}_{ma} \\ \dot{Z}_{ma} \\ \omega_{mxa} \\ \omega_{mya} \\ \omega_{mza} \end{bmatrix} = \begin{bmatrix} n \cdot v_a + (p \times n) \cdot \omega_a \\ o \cdot v_a + (p \times o) \cdot \omega_a \\ a \cdot v_a + (p \times a) \cdot \omega_a \\ n \cdot \omega_a \\ o \cdot \omega_a \\ a \cdot \omega_a \end{bmatrix} \quad \text{----(6)}$$

$$\dot{X}_{md} = \begin{bmatrix} \dot{X}_{md} \\ \dot{Y}_{md} \\ \dot{Z}_{md} \\ \omega_{mxd} \\ \omega_{myd} \\ \omega_{mzd} \end{bmatrix} = \begin{bmatrix} n \cdot v + (p \times n) \cdot f \\ o \cdot v + (p \times o) \cdot f \\ a \cdot v + (p \times a) \cdot f \\ n \cdot f \\ o \cdot f \\ a \cdot f \end{bmatrix} \quad \text{----(7)}$$

$$e = \begin{bmatrix} C(h\phi)(n \cdot \dot{d}_T) + S(h\phi)(o \cdot \dot{d}_T) \\ -S(h\phi)(n \cdot \dot{d}_T) + C(h\phi)(o \cdot \dot{d}_T) \\ a \cdot \dot{d}_T \end{bmatrix}$$

$$f = \begin{bmatrix} -[S\psi C(h\phi) - C\psi S(h\phi)]\dot{\theta} \\ [C\psi C(h\phi) + S\psi S(h\phi)]\dot{\theta} \\ \dot{\phi} \end{bmatrix}$$

$$\dot{d}_T = [\dot{X} \ \dot{Y} \ \dot{Z}]^T$$

3. 질량 중심에서의 SERVO

서버의 목적은 위치와 속도 오차를 수정하기 위해 질량 중심에 작용하는 힘과 모멘트를 계산하는 것이다. 질량 중심의 실제 Cartesian 가속도 \dot{X}_{ma} 는

$$\begin{aligned} \dot{X}_{ma}(t) &= -K_v[\dot{X}_{ma}(t) - \dot{X}_{md}(t)] \\ &\quad - K_p[X_{ma}(t) - X_{md}(t)] + \dot{X}_{md}(t) \end{aligned} \quad \text{----(8)}$$

여기서 K_v 와 K_p 는 상수이다. 따라서

$$F_m(t) = M \ddot{X}_{ma}(t) \quad \text{----(9)}$$

이다. 여기서 행렬 M 은

$$M = \begin{bmatrix} m_t & 0 & 0 & 0 & 0 & 0 \\ 0 & m_t & 0 & 0 & 0 & 0 \\ 0 & 0 & m_t & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{yy} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{zz} \end{bmatrix}$$

이고, m_t 는 end-effector에서의 총 질량이며 I_{xx}, I_{yy}, I_{zz} 는 질량 중심의 기본속에서 작용하는 관성이다. 여기서 (6) 식을 라플라스 변환하고 오차 응답에 대한 정의,

$$X_{me}(S) = X_{ma}(S) - X_{md}(S) \quad \text{----(10)}$$

를 이용하면 (6) 식은 다음과 같이 되며,

$$(S^2 + SK_v + K_p)X_{me}(S) = 0 \quad \text{----(11)}$$

$K_p = \omega_n^2$ 으로 정의하면 서어보 시스템에 대한 조건은 $K_v = 2\zeta(\omega_n)$ 가 된다. 이때, ζ 는 감쇠 계수이며 매니플레이터가 질량을 갖지 않는 이상적인 경우에 이것은 1 이 된다. [1]

4. 매니플레이터의 조인트 토크를 계산한다.

end point 에서의 힘 F_n 을 조인트 토크로 T 로 변환시켜 줘야한다. 이때 힘 F_n 은,

$$F_n = \begin{bmatrix} n \cdot F_{at} \\ o \cdot F_{at} \\ a \cdot F_{at} \\ n \cdot ((F_{at} \times P) + F_{ar}) \\ o \cdot ((F_{at} \times P) + F_{ar}) \\ a \cdot ((F_{at} \times P) + F_{ar}) \end{bmatrix} \quad \text{----(12)}$$

여기서, $F_{at} = [F_x \ F_y \ F_z]^T$

$F_{ar} = [N_x \ N_y \ N_z]^T$

$F_a = [F_x \ F_y \ F_z \ N_x \ N_y \ N_z]^T$

F_x, F_y, F_z : 힘

N_x, N_y, N_z : 모멘트

그리고, 인가되는 조인트 토크 T 는

$$T = J^T F_n \quad \text{----(13)}$$

와 같이 계산되며, 여기서 J^T 는 Jacobian 행렬의 Transpose 이다.

III. 질량 중심의 힘을 수립시키기 위한 제어

만약 질량 중심에서 실제 토크가 원하는 힘 F_m 만큼 얻어진다면 질량 중심은 원하는 궤적을 따라 동작할 것이다. 이상의 작업을 성취하기 위해, Force convergent 제어를 개발한다.

여기서는 관측되는 힘 F_o 를 측정하기 위해 Wrist Force 센서를 사용한다. 이때 wrist 센서에서의 등가 중력 부하에 대한 계산과 wrist 센서에서 인가되는 F_a 를 구하는 방법은 다음과 같다.

1. 질량에 의해 발생하는 Wrist 센서에서의 등가 중력

기본 좌표계에 대한 질량 중심에서의 중력은 $F_g = [0 \ 0 \ -m_e g \ 0 \ 0 \ 0]^T$ 이며 질량 중심의 위치 POSM은,

$$POSM = \begin{bmatrix} n_m & o_m & a_m & p_m \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{----(14)}$$

이다. 만약 벡터 $C_g = [0 \ 0 \ -m_e g]^T$ 라 정의하면 기본 좌표계에서의 등가 중력은

$$F_{go} = \begin{bmatrix} C_g \\ p_m \times C_g \end{bmatrix} \quad \text{----(15)}$$

이고 Wrist 센서에서의 등가 중력 F_{gw} 는

$$F_{gw} = \begin{bmatrix} n_1 \cdot C_g \\ o_1 \cdot C_g \\ a_1 \cdot C_g \\ n_1 \cdot ((p_m - p_1) \times C_g) \\ o_1 \cdot ((p_m - p_1) \times C_g) \\ a_1 \cdot ((p_m - p_1) \times C_g) \end{bmatrix} \quad \text{----(16)}$$

여기서 n_1, o_1, a_1, p_1 은 조인트 변환과 Wrist 변환의 결합으로 이루어진 행렬의 새 벡터 일이다. 따라서 초기에 인가되는 힘은 $F_a(0) = F_{gw}$ 이다.

2. Wrist 센서에서 인가되는 힘 F_{aw} 의 예측
알 질에서 서어보 오차를 줄이기 위해 질량 중심에 인가되는 F_{aw} 를 예측하는 Force Convergent 방법을 개발한다. 이러한 F_{aw} 와 측정되는 힘 F_o 간의 관계는 다음과 같다.

$$F_o = f(F_{aw}) \quad \text{----(17)}$$

이 방법은 F_o 가 일고자 하는 힘 F_{mw} 에 수렴하도록 하는 F_{aw} 를 결정하는 것이다.

3. 질량 중심에서 원하는 힘 F_m 의 보상
매니플레이터의 미지 유효 질량과 관성때문에 $\tau = 1$ 일 경우, 서어보 시스템의 응답은 감쇠 진동한다. 따라서 감쇠 계수 τ 를 증가시킴으로써 매니플레이터 질량에 대해 보상을 해주는 근사화 방법을 개발한다. τ 의 값은 다음의 관계식에 의해 선택된다.

$$\tau = (m_a + m_e) / m_e \quad \text{----(18)}$$

여기서, m_a : 매니플레이터의 유효 질량
 m_e : end-effector에서의 총 질량
Stanford Arm의 경우, m_a 의 값은 다음과 같이 근사화된다.

$$m_a = \sum m_i + m_g \quad \text{----(19)}$$

여기서 m_i 는 i 번째 링크의 질량이다.

IV. 알 고 리 즘

Known :

- 1) 원하는 궤적에 대한 매니플레이터의 CONTROL POINT
- 2) 서어보 표본 시간 : Δt
- 3) 물체의 질량, 관성 그리고 질량 중심
- 4) 힘 수렴(FORCE CONVERGENT) 단계의 수 : N

- 단계 1 : m_e 와 MASS의 계산
여기서 MASS는 END POINT의 좌표와 등 질량의 질량 중심에 대한 기본축 간의 등차 변환
- 단계 2 : τ, K_p, K_v 의 계산
- 단계 3 : $t = 0$ 의 경우에 대한 $F_{aw}(0) = F_{gw}$ 의 보상과 $\dot{X}_{md}(0) = 0$ 의 정의
- 단계 4 : 각 서어보 시간마다 다음 단계를 실행한다.
- 단계 5 : 실제 조인트 위치를 측정하고 $X_{me}(t)$ 를 계산한다.
- 단계 6 : 실제 조인트 속도를 측정하고 $\dot{X}_{me}(t)$ 를 계산한다.
- 단계 7 : 원하는 속도 $\dot{X}_{md}(t + \Delta t)$ 와 원하는 가속도 $\ddot{X}_{md}(t)$ 를 구한다.
- 단계 8 : $F_m(t)$ 를 계산하고 이를 $F_{mw}(t)$ 로 변환한다.
- 단계 9 : N 단계에 대한 FORCE-COMPARE-ADD를 수행하고 단계 4로 가라.

V. 시 룰 태 이 션

본 논문에서 제시된 제어 기법의 보상 능력을 알아 보기 위하여 이를 JPL STANFORD ARM에 적용하여 시뮬레이션해 보았다.

여기서 CONTROL POINT와 부하의 질량 중심은 임의로 지정하였으며 물체는 $4 \times 4 \times 4$ (inch)의 입방체이다. 이때, 측정되는 힘 $F_o(t)$ 는 다음 식에서 추정할 수 있다.

$$F_o(t) = M \left(\frac{\dot{X}_{ma}(t) - \dot{X}_{ma}(t - \Delta S)}{\Delta S} \right) \quad \text{----(20)}$$

여기서, ΔS 는 Wrist 센서의 측정 시간이며, 본 논문에서는 $\Delta S = 1$ msec로 지정하였다. CONTROL POINT의 원하는 Set Point 위치 순서는

$$POSC1 = \begin{bmatrix} 0 & 1 & 0 & 30 \\ 0.6 & 0 & 0.8 & 10 \\ 0.8 & 0 & -0.6 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$POSC2 = \begin{bmatrix} 0 & 1 & 0 & 30 \\ 0.6 & 0 & 0.8 & 10 \\ 0.8 & 0 & -0.6 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$POSC3 = \begin{bmatrix} 0 & 1 & 0 & 30 \\ -0.8 & 0 & -0.6 & -10 \\ -0.6 & 0 & -0.8 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$POSC4 = \begin{bmatrix} 0 & 1 & 0 & 30 \\ 0.8 & 0 & -0.6 & -10 \\ -0.6 & 0 & -0.8 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

이며, 이에 따른 궤적은 [1]에 나와있는 방법을 이용하여 계산한다. 그리고 Force Convergent 방법에서의 N 은 N = 1 과 N = 2 로 선택하였다.

이상의 조건을 근거로 하여, 본 논문에서는 물체의 질량이 다른 경우에 대해 제어 기법을 적용하였다. 그림 2 ~ 4는 매니플레이터가 부하를 갖지 않을 경우에 대한 Control Point 에서의 오차를 나타낸다. 이때, Force Convergent는 Step수 N 에 의해 좌우된다. N = 2 일 경우, 오차 응답은 Zero에 수렴하며, N = 1 일 경우보다 오차 응답이 훨씬 작다. 그리고 물체의 무게가 클수록 오차는 작아진다. 여기서 사용된 알고리즘은 Fo를 Fm으로 수렴하게끔 제어해 준다.

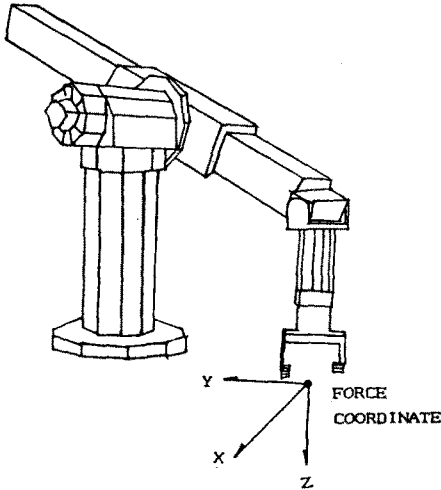


그림 1. Stanford Arm 의 형태

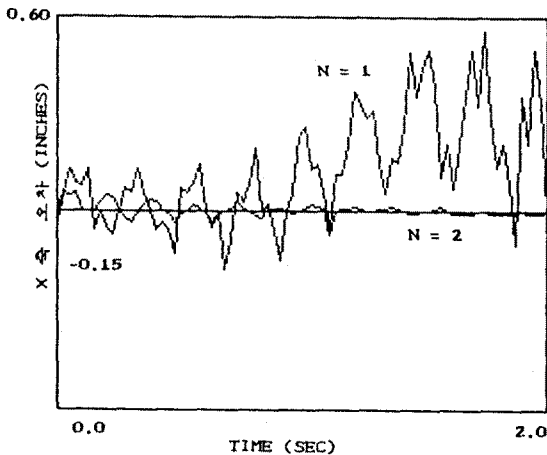


그림 2. NO-LOAD 의 경우, CONTROL POINT 의 X 속을 따라 발생하는 위치 오차

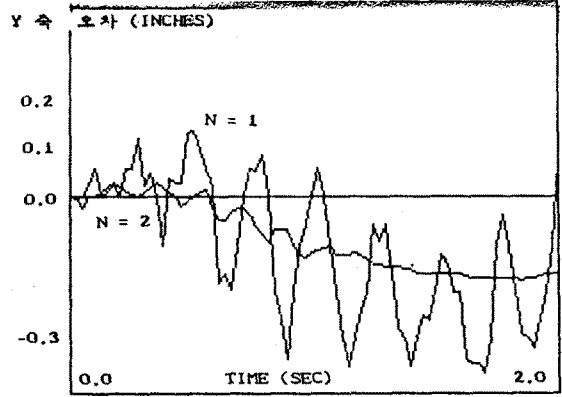


그림 3. NO-LOAD 의 경우, CONTROL POINT 의 Y 속을 따라 발생하는 위치 오차

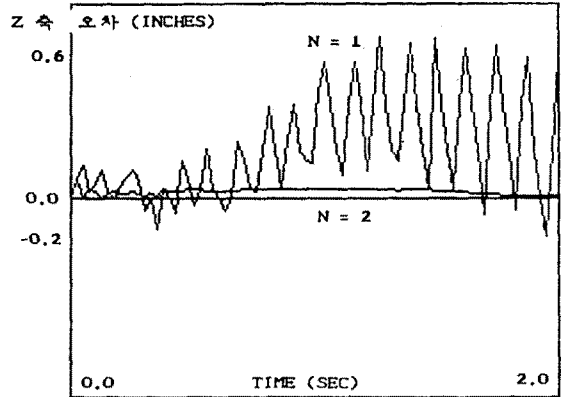


그림 4. NO-LOAD 의 경우, CONTROL POINT 의 Z 속을 따라 발생하는 위치 오차

VI. 결 과

이상의 시뮬레이션 결과를 통해서, 제어 기법은 어떠한 종류의 부하에 대해서도 정확하게 궤적을 따라 이동할 수 있게끔 제어해 주며 매니플레이터의 중력 보오크, 시스템 형태의 변화에 대해 자동적으로 보상하는 Cartesian 경로 제어를 제시해 준다는 것을 알 수 있었다.

이 제어 기법은 6 x n Jacobian 행렬의 전치 형태를 사용하기 때문에, 어떠한 자유도를 갖는 매니플레이터에 대해서도 적용시킬 수 있다.

참 고 문 헌

[1] Paul, R., "Manipulator Cartesian Path Control", IEEE Trans. Syst., Man, Cybern., Nov. 1979.
 [2] Whitney, D. E., "The Mathematics of Coordinated Control of Prostheses Arm and Manipulators", J. of Dynamic Systems, Meas., and Contr., Trans. of ASME, DEC. 1972.
 [3] Paul, R., B. Shimano, "Compliance and Control", Joint A.C. Conf., SAN FRANCISCO, 1976.