

스카라형 로봇의 동작에 관한 2차 PID 제어기의 마이크로프로세서 구현

박 일 영 홍 석 교
아주대학교 제어공학과

Microprocessor Implementation of Secondary PID Controller
for Motion of SCARA Type Robot

Il Young Park Suk Kyo Hong
DEPT. OF CONTORL ENG. AJOU UNIV.

ABSTRACT

In this paper, the controllers for the motion of the 2-joint manipulator design two stages: (1) a primary controller that under ideal conditions makes the end-effector track the desired trajectory; (2) a secondary controller that compensates for undesirable deviations of the motion from the disired trajectory caused by external and/or internal disturbances. The secondary contoller is applied to PID control algorithm, and the controllers is actually designed using IBM-PC/AT and 8096 single chip microprocessor.

1. 서 론

로봇 매니플레이터가 요구된 관절 운동을 실행하기 위해서는 구동기(actuator)에 힘 또는 토크 명령을 전달하는 시스템이 필요하다. 매니플레이터의 제어 시스템은 각 관절의 센서로부터 입력되는 궤환신호(feedback signal)를 이용함으로써 힘/토크 명령을 연속적 혹은 간헐적으로 계산한다.[1] 그런데, 매니플레이터는 각 관절의 동특성 변수가 링크 상호간에 결합(coupling)되고 부하에 따라 파라메터도 변하는 비선형 시스템으로 이루어져 있다. 즉, 매니플레이터의 동특성 방정식은 관절들간에 서로 결합된 비선형 시변함수(Joint Coupled Nonlinear Time Varying Function)이기 때문에 로봇의 위치를 정확하게 제어하고자 할 때에는 어려움이 따른다.[2]

본 논문에서는 독립 관절 변수 제어(Independent Joint Variable Control)에 기초하여 로봇 매니플레이터의 동작에 대한 제어기를 설계하였는데, 제어기는 이상적인 조건에서 요구된 궤적을 추적하는 1차적 제어기와 외란(disturbance)에 의해 야기되는 요구된 궤적으로부터의 오차를 보상해주는 2차적 제어기로 구성하였다.[3] 특히 2차적 제어는 PID(Proportional Integral Derivation : 비례 적분 미분) 제어 알고리즘을 이용하였고, 이 제어기를 실제 설계함에 있어서 싱글칩 마이크로프로세서인 8096을

사용하였고, 제어 대상으로는 Rhino-SCARA 형 2축 로봇를 사용하였다. 2축 매니플레이터의 직선 궤적에 대한 시뮬레이션 및 실험을 해보았다.

2. 매니플레이터의 수학적 모델링

제어의 관점에서 볼 때 로봇 매니플레이터는 고도의 비선형적으로 결합된 다변수 시스템으로 운동특성은 비선형 시변 2계 미분 방정식으로 표현된다. N개의 관절을 갖는 직렬 링크(serial link) 매니플레이터의 전체 동적 모델은 관절들의 상호 의존적인 영향을 효과를 보여주는 Lagrange 형식으로 나타낸다.[2] 동적 모델의 Coliolis 부분은 관절들 사이의 결합된 미분 방정식 모델을 의미한다.

그러나, 어떤 조건하에서는 결합된 부분이 모델 방정식에서 부분적으로 보상되거나 무시될 수도 있다.[3] 이것은 다른 관절의 상호작용(interaction) 영향없이 이끌어진 관절 운동의 방정식을 나타낸다. 따라서 관절 i에 대한 운동 방정식은 관절 변수 q_i 와 관절에 입력되는 τ_i 만을 포함한다. 만약 출력이 관절의 위치를 나타내는 모델이라면 그 모델에 대하여 설계된 제어를 보통 독립 관절 변수 제어(Independent Joint Variable Control)라 한다.[2]

일반적인 로봇 매니플레이터의 동적모델은 다음의 2차 비선형 미분 방정식으로 표현된다.

$$\tau(t) = D(q(t))\dot{q}(t) + h(q(t), \dot{q}(t)) + c(q) \quad (1)$$

식 (1)을 가지고 SCARA형 2축 로봇의 동적 방정식을 유도하면 다음과 같다.

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} \frac{m_1 l^2}{3} + \frac{4m_2 l^2}{3} + m_2 \cos^2 \theta_2 & \frac{-m_2 l^2}{3} + \frac{m_2 l^2 \cos \theta_2}{2} \\ \frac{m_2 l^2}{3} + \frac{-m_2 l^2 \cos^2 \theta_2}{2} & \frac{m_2 l^2}{3} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix}$$

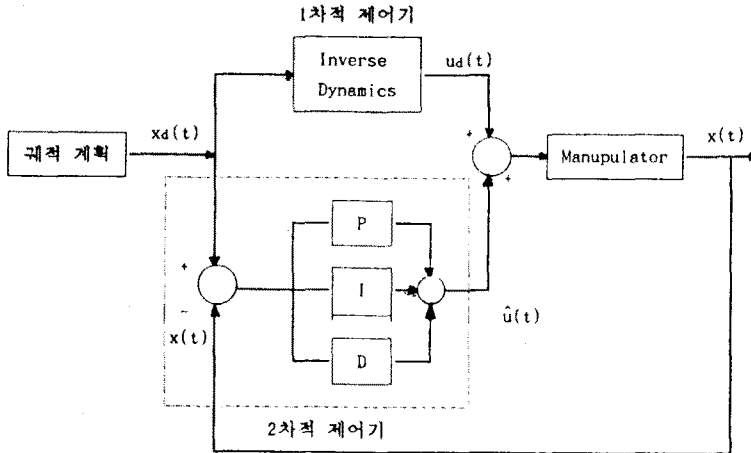


그림 1. 제어기의 블록 다이어그램

$$\begin{bmatrix} \frac{m_2 \sin q_2 l^2 q_2^2 - m_2 \sin q_2 l^2 q_1 q_2}{2} \\ - \frac{m_2 \sin q_2 l^2 q_1^2}{2} \end{bmatrix} \quad (2)$$

- $\tau(t)$: 관절에 가해지는 토크 벡터
- $q(t)$: 관절 변수 벡터
- $\dot{q}(t)$: 관절 속도
- $\ddot{q}(t)$: 관절 가속도
- $D(q)$: symmetric positive definite 관성 행렬
- $h(q)$: Coriolis력 및 원심력 벡터
- $c(q)$: 중력 하중 벡터

대부분의 매니플레이터의 동적 모델은 비선형적이고 계산적으로 복잡하다. 제어기의 일반적인 모델은 선형 시불변 동적 모델에 기초하고 있다.[3] 따라서 매니플레이터에 대한 동적모델의 선형화는 식 (1)에 feedback-feedforward 선형화 방법을 적용하여 다시 표현할 수 있다.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2(t) \\ f_2(x) \end{bmatrix} + \begin{bmatrix} 0 \\ B_2(x) \end{bmatrix} u(t) \quad (3)$$

여기서

- $u(t)$: 입력 벡터
- $f_2(x)$: 독립변수의 연속함수 벡터
- $B_2(x)=D^{-1}$: 독립변수의 연속함수 행렬
- $x(t)=[x_1(t)', x_2(t)']'$ 는 2-dimensional 상태 벡터
- 즉, $x_1(t)=[q_1(t), q_2(t)]'$ 관절의 위치벡터
- $x_2(t)=[\dot{q}_1(t), \dot{q}_2(t)]'$ 관절의 속도벡터

3. 제어기의 설계

매니플레이터의 동작 제어에 대한 문제는 엔드 이펙터의 이동하는 점들이 요구되어지는 궤적 즉, 기준치(reference value)를 추적(tracking)하여야 하는 서어보 시스템에 관한 것이다. 매니플레이터의 입력을 관절 구동기에 의해 토크가 발생되고, 제어된 출력은 제어가 관절 공간에서 수행될 때의 값인 위치, 속도, 가속도가 된다. 관절의 실제 위치와 속도는 엔코더에 의해 측정되고, 가속도는 소프트웨어 프로그램에 의해 계산된다. 이들 변수는 폐루프(feedback loop)를 사용하여 위치, 속도, 가속도 오차의 형태와 원하는 값들로 비교된다.

매니플레이터의 동작에 대한 제어기의 설계는 2가지 단계로 구성될 수 있다.

- ① 엔드 이펙터가 요구된 궤적을 추적한다는 이상적인 조건하에서 설계 (1차적 제어기)
- ② 선형화된 모델에 기초하여, 외부 또는 모델 자체의 외란에 의해 발생하는 요구된 궤적에서의 오차를 보상해 주는 제어기 설계 (2차적 제어기)

매니플레이터 각 관절의 토크 입력을 그림 1과 같이 제어기를 구성하면

$$\tau(t) = \tau^d + k_p(\dot{q}^d - \dot{q}) + k_p(q^d - q) + k_i \int (q^d - q) dt \quad (4)$$

$$u(t) = u^d(t) + \hat{u}(t) \quad (5)$$

로 유도된다.

(1). 1차적 제어기

비선형 모델에 feedback-feedforward 선형화 방법으로 선형화하는 식(1)을 이용하여 원하는 궤적에 대한 방정식을 구하면

$$\ddot{x}^d = f_2(x^d) + B_2(x^d)u^d(t) \quad (6)$$

이다. 여기서 $B_2^{-1}(x)$ 는 관성행렬(D(q))이므로, 모든 시간에 대해서 존재하고 $u^d(t)$ 에 의해 풀 수 있다. 이 식은 매니플레이터의 역동적(inverse dynamics : $x^d(t) \rightarrow u^d(t)$)를 의미한다.

실제 시스템에서는 IBM-PC에서 계산되어지며, $x^d(t)$ 에 대한 궤적 계획 또한 IBM-PC에서 행해진다.

(2). 2차적 제어기

동적모델의 파라미터 값들이 정확하지 않기 때문에 1차적 제어기에서 비선형성의 제거는 완전하게 수행되지 못한다. 따라서 2차적 제어기에서 외란에 따른 오차를 보상해 주는 PID 제어 알고리즘을 수행한다.

오차 보상에 대한 PID 제어 부분은 식 (4)에서

$$\tau^d(t) = K_p(q^d - q) + K_v(\dot{q}^d - \dot{q}) + K_i \int (q^d - q) dt \quad (7)$$

를 계산한다.

이 제어 알고리즘은 8096(single-chip microprocessor)에서 계산되어지며, 이 제어기에서 계산된 $\tau^d(t)$ 와 PC에서 계산되어 전송된 $\tau^d(t)$ 를 비교하여 매니플레이터에 힘/토크 명령을 준다.

(3). 궤적 계획 (Trajectory planning)

매니플레이터의 이동경로에 대한 기준궤적은 LSPB (Linear Segments with Parabolic Blends) 알고리즘에 의해 설정한다.

$$q^{d_i}(t) = \begin{cases} q_0 + \frac{a}{2}t^2 & 0 < t \leq t_b \\ \frac{q_f + q_0 - Vt}{2} + Vt & t_b < t \leq t_b - t_f \\ \frac{at_f^2}{2} + at_f t - \frac{a}{2}t^2 & t_f - t_b < t \leq t_f \end{cases}$$

4. 전체 시스템의 구성

본 논문에서 구성한 제어 시스템은 크게 로봇트 팔, 제어부(1차, 2차), 구동부, 센서부로 나눈다.

● 로봇트 팔 : Rhino SCARA형 2-축 로봇트

- ① 길이 : $l_1 = l_2 = 0.23m$
- ② 무게 : $m_1 = 2.55kg, m_2 = 2.96kg$
- ③ 사용 모터 : DC 서어보 모터 정격 12V
- ④ 최대속도 : 구동 전압 16[V] dc 일 때
축 1 = 29.30°/sec
축 2 = 51.78°/sec
- ⑤ Encoder 분해능 : 축 1 = 0.12°
축 2 = 0.23°

● 제어부

- ① IBM-PC : 1차적 제어부로서 궤적 계획 및 그 궤적에 대한 요구된 힘/토크를 계산하여, 계산한 네

이터를 각 샘플링 시간마다 2차적 제어부에 전송.

- ② 8096 single-chip microprocessor : 1차적 제어부로부터 전송받은 요구된 힘/토크 데이터를 측정된 위치, 속도, 가속도를 가지고 계산한 실제 힘/토크와 비교, 속도 명령에 비례하는 PWM 펄스를 발생시키고, 내부 Timer를 써서 엔코더 펄스를 카운트하여 매니플레이터의 각 관절의 움직임 변위를 계산.

- 센서부 : 모터 1회전당 출력되는 엔코더 펄스의 수가 6개이므로, 4체배하여 1회전당 24개의 엔코더 펄스를 발생시킨. 즉, 엔코더 1펄스당 축 1은 0.04°, 축 2는 0.0575°가 됨.

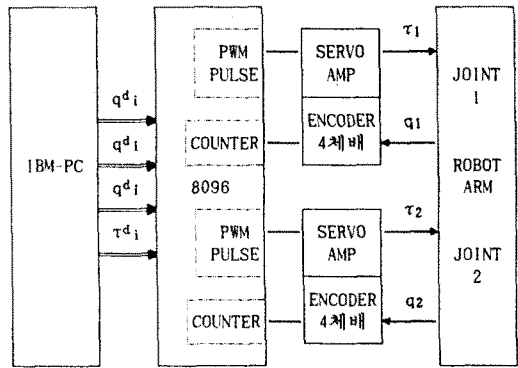


그림 2. 시스템의 하드웨어 구성

5. 시뮬레이션

시뮬레이션은 그림 1의 제어기를 구성으로 IBM-PC에서 C언어로 프로그램 하였다.

원하는 궤적은 매니플레이터 각각의 관절이 4초간 경로에 따라 가속, 감속을 하며 60°를 이동하는 동작으로 설정하였다. 로봇트의 동작 환경은 무부하 상태로 하고 마찰력은 속도에 비례하게 설정하였다. 그리고 미리 정한 K_p, K_i, K_d 의 값으로 시뮬레이션 해보았다.

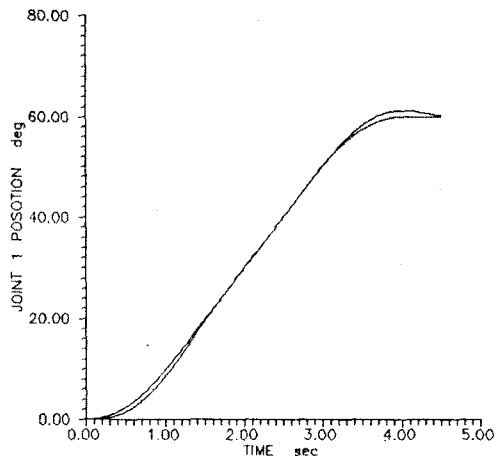


그림 3. 축 1의 기준궤적과 시뮬레이션 결과

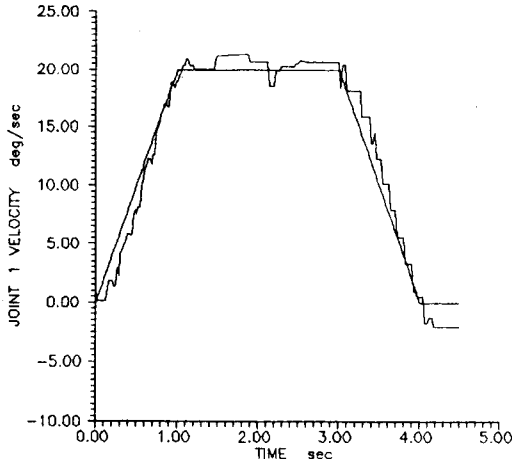


그림 4. 축 1의 기준속도궤적과 시뮬레이션 결과

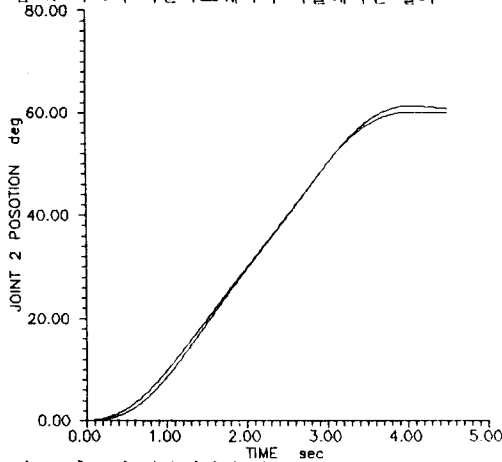


그림 5. 축 2의 기준궤적과 시뮬레이션 결과

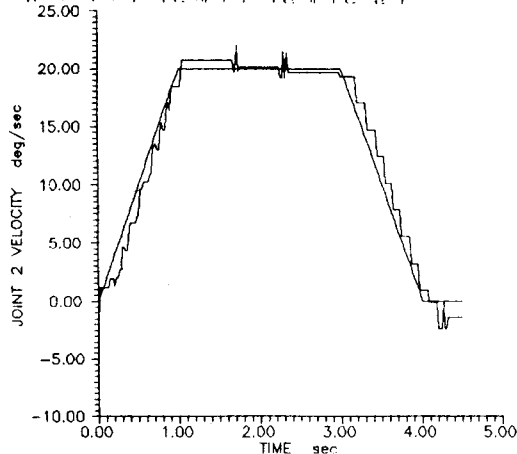


그림 6. 축 2의 기준속도궤적과 시뮬레이션 결과

6. 결 론

본 논문은 로봇트 매니플레이터의 제어기 설계를 궤적 계획과 역동역학(inverse dynamics) 계산을 하는 1차적 제어기와 PID 제어 알고리즘인 2차적 제어기로 구성하여 2개의 시스템으로 나누어 설계함으로써 각 제어기의 계산

에 대한 부담을 덜어주었다.

그러나 PID 알고리즘을 비롯한 여러 알고리즘이 무부하시 시스템의 과도 응답 특성을 개선하기 위한 것이어서 실제 운전시 필수적으로 따르는 부하 문제를 고려하지 않게 된다. 따라서 부하의 변화나 그밖의 동적 파라미터들의 변화에 따른 적응 제어 알고리즘으로 개선하는 것이 이 논문의 과제이다.

참 고 문 헌

- [1] K.S.Fu, R.C.Gonzalez, and C.S.G.Lee, "ROBOTICS : Control, Sensing, Vision, and Intelligence," McGraw-Hill Book Company, 1988.
- [2] Mark W.Spong, M.Vidyasagar, "ROBOT DYNAMICS AND CONTROL," John Wiley and Sons, Inc. 1989.
- [3] Antti J. Koivo, "Fundamentals fo Control of Robotic Manipulators," John Wiley & Sons, Inc. 1989.
- [4] Robert J.Schilling, "FUNDAMENTALS OF ROBOTICS : Analysis and Control," Prentice-Hall, Inc. 1990.
- [5] J.Y.S.Luh, M.W.Walker, R.P.C.Paul, "On-line Computational Scheme for Mechanical Manipulators" ASME J. Dyn. Syst. Meas. Contr., vol. 102, no. 2, pp.69-76, 1980.
- [6] H.Ari and S.Tachi, "Position Control System of a Two Degree of Freedom Manipulator with a Passive joint," IEEE Trans. Electr. vol 38, No. 5, February, 1991.
- [7] H.Zang, G.Troot, R.p.Paul, "Minimum delay PID control of Interpolated Joint Trajectories of Robot Manipulators," IEEE Trans. Electr. vol 37, No.5, October, 1990.
- [8] P.K.Khosla, T.Kanade, "Experimental Evaluation of Nonlinear Feedback and Feedforward Control Schemes for Manipulators," Int. J. Robotics Research, vol 7, No 1, February 1988.
- [9] 박경희, 홍석교, "로봇트 매니플레이터를 위한 강인한 제어기 설계에 관한 연구" 전기학회 추계종합학술대회 pp 450-455, 1989.11.