

로봇트 경로 제어를 위한 학습 제어 방법
A Learning Control Method
for Path-Control of Robot Manipulators

오 상 륙^o
변 중 남
서 일 흥

한국과학기술원
한국과학기술원
한양대학교

I. 서 론

산업 현장에서의 활용도가 점점 증가하는 로봇트를 아크 용접이나 실링 (Sealing) 작업 등과 같은 작업 경로가 중요한 작업에 응용할 경우, 로봇트는 주어진 연속 경로를 전 구간 동안 사용자가 제시한 경로 오차 한계 내에서 빠른 동작으로 추종할 수 있는 기능을 갖고 있어야 한다. 그러나 복잡한 비선형 특성을 갖는 로봇트를 정확하게 모델링하기는 쉬운 일이 아니며 벡 래쉬, 마찰력 및 비 강체 특성 등은 모델링 자체가 곤란하여, 부정확한 모델 및 미지의 파라미터를 갖는 모델로부터 로봇트의 정확한 연속 경로 제어는 상당히 어려운 문제이다. 이를 구체적으로 표현하면, 미지의 파라미터를 갖는 로봇트의 다이내믹스가 다음과 같이 주어졌다고 하자.

$$\dot{x}(t) = f(x(t), u(t); \lambda), \quad x(t_0) = \xi^0 \quad \dots \dots \quad (1)$$

(1)식에서 x , u 는 각각 n 차원 상태 벡터, m 차원 입력 벡터이고, λ 는 미지의 파라미터를 나타낸다. 전술한 바, 연속경로 제어 문제는 다음과 같이 쓸 수 있다.

(문제 1) 원하는 상태 궤적 $x^D(t)$, $0 \leq t \leq T$ 와 오차 한계 $\epsilon > 0$ 가 주어졌다고 하자. 미지의 파라미터를 갖는 시스템 (1)에 대해서

$$p(x) = \|x(t) - x^D(t)\| \leq \epsilon, \quad 0 \leq t \leq T \quad \dots \dots \quad (2)$$

를 만족하도록 입력 함수 $u(t)$, $0 \leq t \leq T$, 를 구하라.

(2)식에서 $\|\cdot\|$ 는 n 차원 maximum-norm 을 나타낸다.

(문제 1)을 풀기 위한 한 방법으로 적응 제어 기법을 생각할 수 있다. 그러나 기존의 적응 제어 기법으로는 점근 수렴성만이 보장될 뿐 초기 혹은 간헐적인 동작은 제어되지 않기 때문에 (2)로 주어지는 상태 제약 조건을 만족할 수 없게 된다[1].

이러한 문제를 풀기 위하여 본 논문에서는 반복 학습 제어 방법을 제시한다. 지금까지 몇 개의 반복 학습제어 방법이 발표되고 있는데, Raibert [2]는 매개변수 기법(parameterization technique)을 이용한 학습 제어방법을 제시하였다.

그러나 이 방법에서는 랭크(rank)가 모자라는 행렬의 역행렬을 빈번히 계산해야 하는 어려움 때문에

응용에 상당한 제약을 받고 있다. [3]에서는 Arimoto 와 그의 연구진이 간단한 구조를 갖는 반복 학습 제어 방법이 제시되었다. 그러나 수렴성이 보장되는 제어기의 이득을 정하기 위해서 시스템의 다이내믹스 특성을 정확히 알아야 한다는 단점을 갖고 있다. 또한 이 방법에서는 비선형 시스템을 원하는 궤적 주위에서 선형화하기 때문에, 미지의 파라미터를 갖는 시스템에서 흔히 발생할 수 있는 큰 외란이 발생할 경우에는 효과적으로 적용할 수 없는 문제점을 안고 있다.

본 논문에서는 (문제 1)을 풀기 위하여 선형화 과정이 필요치 않는 새로운 형태의 반복 학습 제어 방법을 제안하기로 한다.

II. 학습 제어 알고리즘

(1)식으로 주어지는 로보트 다이내믹스를 다음과 같이 모델링하였다고 하자.

$$\begin{aligned} \dot{x}^M(t) &= f(x^M(t), u(t); \lambda^M), \\ x^M(0) &= \xi^0 \end{aligned} \quad (3)$$

λ^M 은 λ 에 대한 모델링을 나타낸다. (3)식으로 부터 제어 입력 $u(t)$ 를

$$u(t) = g(x^M(t), \dot{x}^M(t); \lambda^M) \quad (4)$$

와 같이 나타낼 수 있다고 하자. 여기서 $u(t)$ 는

$$\begin{aligned} x^M(t) &= \xi^{0M} + \int_0^t f(x^M(\tau), g(x^M(\tau), \\ &\quad \dot{x}^M(\tau); \lambda^M); \lambda^M) d\tau \end{aligned} \quad (5)$$

를 만족하며 역 시스템 모델이라고 한다.

앞으로 suffix-indexing 을 사용하여 반복 숫자를 나타내기로 한다. 즉, $x_k^M(t)$ 는 k 번째 반복 동작

의 t 에서의 플랜트 상태 벡터의 값을 나타낸다.

이제 (1)식의 비선형 시스템에 대하여 (문제1)에 대한 해로서 다음과 같은 반복 학습 제어 방법을 제안한다.

(알고리즘 1) : 상태 궤적 오차 한계 $\epsilon > 0$ 와

초기 시행시의 입력 $u_0(t)$, $0 \leq t \leq T$, 가 주어졌다고 하자 또한 초기의 잠정 기준 벡터

$$\begin{aligned} x_0^M(t) \text{가 } x_0^M(t) = x_0(t) \text{ 로 주어졌다고 하자.} \\ k = 0 \text{ 으로 놓는다.} \end{aligned}$$

스텝 1 : $e_k(t) = x^D(t) - x_k(t) \dots (6)$

을 계산한다. 여기서 $x_k(t)$ 는

$$x_k(t) = \xi^0 + \int_0^t f(x_k(\tau), u_k(\tau); \lambda) d\tau \dots (7)$$

로 표현되는 (1)식의 해로서 측정에 의해 구해진다.

스텝 2 : $k+1$ 번째의 잠정 기준 벡터 $x_{k+1}^M(t)$ 를

$$x_{k+1}^M(t) = x_k^M(t) + S e_k(t) \dots (8)$$

와 같이 구한다.

스텝 3 : $k+1$ 번째의 입력 $u_{k+1}(t)$ 를

$$u_{k+1}(t) = g(x_{k+1}^M(t), \dot{x}_{k+1}^M(t); \lambda^M) \dots (9)$$

와 같이 계산한다.

스텝 4 : $\|e_k(t)\| \leq \epsilon$ 이면 스톱하고

아니면 $K = k+1$ 로 하여 스텝 1으로 돌아간다.

여기서 S 는 $n \times n$ 상수 가중 행렬이다. 이 알고리즘은 모델 연산제어 (model algorithm control, [6])의 특수한 경우로 시간 축이 아닌 반복 동작 축으로 확장하여 사용한 것이다.

제안된 학습제어 방법의 유효성을 보이기 위하여

그림 1과 같은 2개의 링크로 구성된 로봇트 팔에 대하여 컴퓨터 시뮬레이션을 행하였다.

III. 응용 예제 : 2 링크 로봇트 매니퓰레이터
회전 조인트들로 구성된 2 링크 로봇트의 다이내믹스는 다음과 같이 쓸 수 있다.

$$\begin{bmatrix} \tau^1 \\ \tau^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{3}m_1\ell^2 + \frac{4}{3}m_2\ell^2 + m_2C_2\ell^2 \\ \frac{1}{3}m_2\ell^2 + \frac{1}{2}m_2C_2\ell^2 \\ \frac{1}{3}m_2\ell^2 + \frac{1}{2}m_2C_2\ell^2 \\ \frac{1}{3}m_2\ell^2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}^1 \\ \ddot{\theta}^2 \end{bmatrix} + \begin{bmatrix} -\frac{1}{2}m_2S_2\ell^2\dot{\theta}^2 \\ \frac{1}{2}m_2S_2\ell^2\dot{\theta}^1 \\ -\frac{1}{2}m_2g\ell C_1 - \frac{1}{2}m_2g\ell C_{12} - m_2g\ell C_1 \\ -\frac{1}{2}m_2g\ell C_{12} \end{bmatrix} \dots (1c)$$

여기서 C_i , S_i 와 C_{ij} 는 각각 $\cos \theta^i$, $\sin \theta^i$, $\cos(\theta^i + \theta^j)$ 이다.

(10)식에서 θ^1 과 θ^2 는 조인트 변수이고, 각 링크의 무게와 길이를 표 1에 보였다. $t \in [0, 1]$ 에 대하여 원하는 궤적과 오차 한계 ϵ 를

$$\begin{aligned} \theta^i{}^D(t) &= -\frac{5}{3}t^3 + \frac{5}{2}t^2, \text{ for } i=1,2 \\ \dot{\theta}^i{}^D(t) &= -5t^2 + 5t, \text{ for } i=1,2 \end{aligned}$$

$\epsilon = 1.0[\text{Degree}] = 0.017[\text{Radian}]$ 으로 주어졌다. 식(6) - 식(9)로 주어진 알고리즘을 적용하기 위해서 모델링을 해야 하는데, 완전한 모델링의 어려움을 고려하여 본 논문에서는 다음과 같이 관성력에 해당하는 항만 모델링하였다고 가정하였다. 즉

$$\ddot{\theta}^M(t) = D^{-1}(\theta^M; \lambda^M) \tau \dots (11)$$

이 모델링으로 부터 역 시스템 모델을 구하면

$$\tau = D(\theta^M; \lambda^M) \ddot{\theta}^M \dots (12)$$

으로 주어진다. 따라서 (12)식으로 표현된 역 시스템 모델을 이용하여 매 반복시 입력을 계산하여 사용하였다. 여기서 미지의 파라미터는 링크의 무게와 길이로 가정하여 실제 값과 모델링된 값을 표 1에 보였다. 첫번 시행시의 입력과 가중 행렬 S 는 다음과 같이 사용하였다.

$$\tau_o(t) = D(\theta^D; \lambda^M) \ddot{\theta}^D(t), \quad 0 \leq t \leq 1$$

$$S = \begin{bmatrix} 0.95 & 0 \\ 0 & 0.95 \end{bmatrix}$$

이상으로 부터 매 반복시 위치의 궤적을 원하는 궤적과 비교하여 그림 2에 보였다. 그림 2에서 볼 수 있듯이, 표 1에 보인 바 파라미터의 모델링 차이가 큰 경우에도 또한 식(12)와 같이 실제 시스템과 상당히 틀린 모델링을 이용하였어도 17번 정도의 반복 시행을 하면 원하는 궤적을 주어진 오차 한계 내에서 잘 쫓아가고 있음을 알 수 있다.

IV. 결 론

제안된 알고리즘은 과거 및 현재의 데이터를 이용하여 미래의 동작이 원하는 동작이 되도록 스스로 제어 입력을 만들어 준다는 의미에서 반복 학습 기능을 갖고 있다는 것을 보였고 이를 비선형 2 링크 로봇트 매니퓰레이터에 응용하여 그 유효성을 보였다.

그러나 전체 시스템의 안정도나 제안된 알고리즘의 수렴성은 아직 연구과제로 남아 있다.

참 고 문 헌

1. C.S.G. Lee, M. J. Chung and B. H. Lee, "An Approach of Adaptive Control for Robot Manipulators," Journal of Robotic Systems, vol. 1, No. 1, 1984, pp. 27-57.
2. M.H. Raibert, "Motor Control of Learning by the STate Space Model," Ph.D. Dissertation, Dept. of Psychology, MIT, 1977.
3. S. Arimoto, S. Kawamura and F. Miyazaki, "Bettering Operation of Robots by Learning," Journal of Robotic Systems, vol. 1, No. 2, 1984, pp. 123-140.

표 1. 입력 데이터

Parameter \ Value	True Value	Modelled Value
m_1	2 Kg	1.5 Kg
m_2	2 Kg	1.5 Kg
l_1	0.5 m	0.4 m
l_2	0.5 m	0.4 m

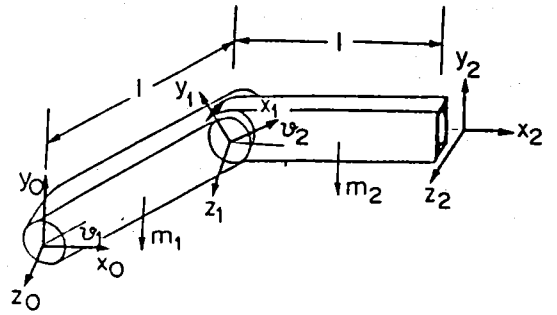
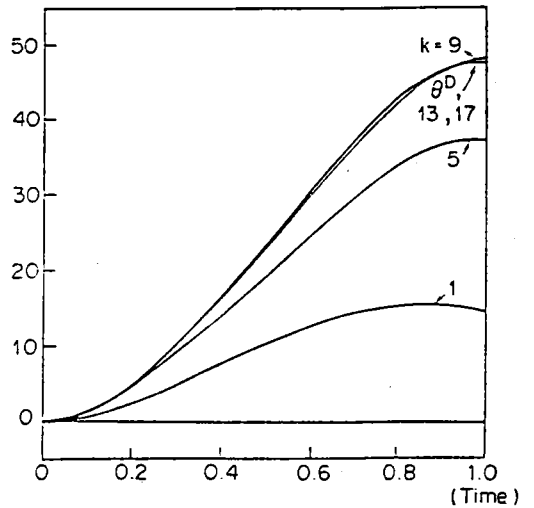


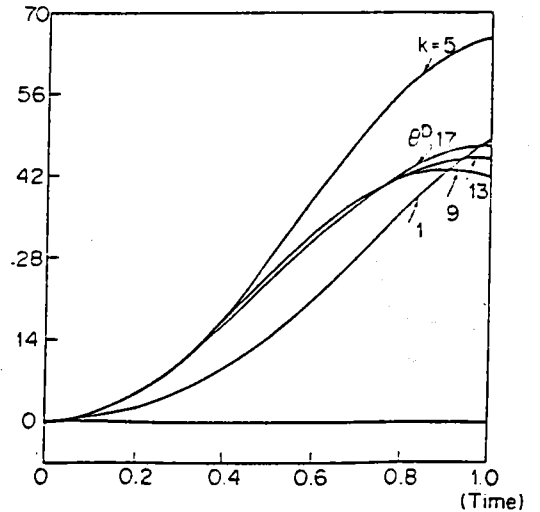
그림 1. 2 링크 로봇트 매니퓰레이터

(Degree)



(a) Joint Angle 1

(Degree)



(b) Joint Angle 2

그림 2. 조인트의 각 궤적