

# 파라미터 추정방법을 이용한 로봇트 반복 학습 제어기의 설계

## Design of an Iterative Learning Robot Controller Using Parameter Estimation

吳 常 綠\* · 黃 東 煥\*\* · 卞 增 男\*\*\* · 徐 一 弘<sup>§</sup>  
 (Sang-Rok Oh · Dong-Hwan Hwang · Zeungnam Bien · Il-Hong Suh)

### 요 약

본 논문에서는 선형 주기성 시변시스템에 대하여 반복 학습제어 방법을 제안하는 바, 시스템의 파라미터를 추정기 및 매 반복시행시 제어압력을 만들어 내는 역 시스템 모델을 이용하였다.

제안된 알고리즘의 수렴성을 보장하는 충분조건을 제시하고 유용성을 보이기 위하여 두개의 수치예제를 결과와 함께 제시하였다. 특히, 제안된 방법은 로봇트머니플레이터의 연속경로제어에 유용하게 사용할 수 있음을 보였다.

**Abstract-** An iterative learning control method is presented for a class of linear periodic systems, in which a parameter estimator of the system together with an inverse system model is utilized to generate the control signal at each iteration. A convergence proof is given and two numerical examples are illustrated to show the validities of the algorithm. In particular, it is shown that the method is useful for the continuous path control of robot manipulators.

### 1. 서 론

엔지니어링 시스템의 지능제어를 위하여 인간과

\*正 會 員 : 韓國科學技術研究院 制御시스템研究室  
 先任研究員 · 工博

\*\*正 會 員 : 韓國科學技術院 電氣電子工學科 博士  
 課程

\*\*\*正 會 員 : 韓國科學技術院 電氣電子工學科 教授 ·  
 工博

§正 會 員 : 漢陽大 工大 電子工學科 副教授 · 工博

接受日字 : 1989年 8月 18日

1次修正 : 1990年 1月 11日

같은 고지능류의 가장 핵심적인 특성중의 하나인 “학습기능”을 시스템제어에 이용하려는 연구가 오래전부터 진행되어 왔다. 특히 로봇틱스의 최근 발전을 토대로 학습기능을 갖는 로봇트 제어기를 설계하는 문제가 새로운 연구과제로 많은 관심을 받고 있다.

[1]에서 정의된 바와 같이 “학습”이란 “시스템이 어떤 일을 수행할 때 같은 일을 다음 시행시에 보다 효과적이고 효율적으로 할 수 있도록 적응하는 시스템에서의 변화”를 말하는 데, 이와 같

은 개념을 보다 좋은 제어를 목적으로 하는 엔지니어링 시스템에 적용하기 위한 연구는 크게 지식 획득(Knowledge Acquisiton) 및 기술숙련(Skill Refinement)의 두 부류로 나눌 수 있고[2], 이 중 특히 반복훈련(Practice)을 통해 기술숙련을 달성하는 학습의 개념이 자동제어 시스템의 많은 분야에 응용될 수 있다고 믿어 왔다.

기존의 적응제어 시스템도 학습 제어 시스템의 한 종류로 생각할 수 있는데 이러한 경우 기술숙련의 목표는 기준치에 도달토록 하는 조절(Regulation)목표가 대부분이다. 그러나 이와 같은 적응제어 시스템의 목표는 한번의 시행으로 달성되어야 하므로 이를, 예를 들어 미지의 다이내믹스를 갖는 로봇트 머니플레이터의 제어에 이용할 경우, 한 점에서 다른 점으로 머니플레이터를 움직이는 위치제어에는 적용할 수 있으나 동작구간 전구간 내에서 주어진 오차한계를 만족하면서 원하는 궤적을 추적하는 경로제어에는 사용할 수 없다. 왜냐하면 기존의 적응제어 시스템은 점근수렴성만을 보장하므로 초기상태나 간헐적인 동작은 제어할 수 없기 때문이다.

학습제어 시스템의 또다른 형태는 반복 학습제어 시스템으로써[5]-[8] 이러한 시스템에서 기술숙련을 위한 학습은 계속되는 반복동작을 통해 이루어진다. 따라서 숙련코자 하는 기술(또는 목표로 하는 성능)이 간단한 형태가 아닌 경우에는 반복학습제어 방법을 훨씬 우수한 제어 방법중의 하나로 생각할 수 있다.

본 논문에서는 미지의 다이내믹스를 갖는 시스템의 효과적인 제어를 위하여 새로운 형태의 반복 학습 제어방법을 제안하기로 하며 이를 아크 용접이나 실링등의 경로추적 과제에 로봇트 머니플레이터를 응용할 경우 요구되는 로봇트 연속경로제어에 적용하고자 한다. 우선 다음과 같은 선형주기성 시스템에 대하여 생각하기로 한다.

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + B(t)u(t) & (1) \\ x(0) &= \xi^0 & (2) \end{aligned}$$

여기서  $x$ 와  $u$ 는 각각  $n \times 1$  상태벡터,  $m \times 1$  입력 벡터이고  $n \times n$  행렬함수  $A(t)$ 와  $n \times m$  행렬함수  $B(t)$ 는 미지의 파라미터를 포함하는 연속함수이며 주기  $T$ 인 주기함수이라고 가정한다. 즉

$$\begin{aligned} A(t+T) &= A(t) & (3) \\ B(t+T) &= B(t) \end{aligned}$$

이제 본 논문에서 다루고자 하는 문제 (P)는 다음과 같다.

문제(P):

원하는 상태궤적  $x^d(t)$ ,  $0 \leq t \leq T$ , 및 오차한계  $\epsilon > 0$ 이 주어졌을 때 (1), (2)식으로 주어지는 시스템의 상태궤적  $x(t)$ 가 다음 (+)식으로 주어지는 상태 부등제약 조건식을 만족하도록 하는 제어 입력함수  $u(t)$ ,  $0 \leq t \leq T$ 를 구하라.

$$P(x(t)) = \|x(t) - x^d(t)\|_{\infty} \leq \epsilon \quad (4)$$

여기서  $\|\cdot\|_{\infty}$ 은  $n$ 차원 최대노름(maximum norm)을 나타낸다.

전술한 바 기존의 적응제어 방법으로는 (4)식으로 주어진 상태 부등제약 조건을 만족할 수 없을 뿐 아니라 일반적으로 미지의 파라미터를 포함하는 시스템에 대하여 (4)식과 같은 상태 부등제약 조건을 만족하는 제어함수를 찾는 조건을 만족할 수 없을 뿐 아니라 일반적으로 미지의 파라미터를 포함하는 시스템에 대하여 (4)식과 같은 상태 부등제약 조건을 만족하는 제어함수를 찾는 것은 상당히 어려운 문제로 알려져 있다. 이러한 문제를 효과적으로 다루기 위하여 제안된 방법중의 하나가 반복학습 제어 방법인데, 구체적으로 Raibert [5]는 파라미터화 기법을 이용한 학습제어 방법을 제시하였으나 이 방법에서는 행렬의 랭크(rank)가 모자라는 행렬의 역행렬을 계산해야 하는 어려움으로 인하여 그 응용에 상당한 제약을 받고 있다. 한편 Arimoto등[6]-[8]은 간단한 형태의 반복학습제어 방법을 제시하고 제시된 방법의 수렴성을 위한 충분조건을 제시한 바 제시된 충분조건을 만족시키기 위하여는 로봇트 다이내믹스를 정확히 알아야 하는 단점을 갖고 있어 실제로 정확히 모델링하기가 상당히 어려운 로봇트 머니플레이터의 경우 효과적으로 사용할 수 없다. 또한 이 방법에서는 매 회수마다 동일한 이득 함수를 사용하므로 학습과정에서 적응성이 없을 뿐만 아니라 최후에 학습된 지식은 입력함수  $u_k(t)$ 에만 국한된다. 그 결과 학습 법칙은 외부 잡음에 의해서 쉽게 영향을 받으며, 모든 입력 함수에 대해서 학습 과정을 반복해야한다.

위의 결점들을 보완하기 위해서 Oh등[14]은 파라미터 추정기를 포함하는 새로운 학습제어 방법을 제안하였는데, 이 방법을 사용하면 수렴속도가 빠를 뿐만 아니라, 외부 잡음들에 보다 강인한 학습 제어기의 구현이 가능하다. 또한 학습된 지식의 관점에서든 입력 함수 뿐만 아니라 시스템의 파라미터를 학습하였다고 할 수 있으므로

더욱 더 많은 지식을 얻을 수 있다. 그러나 [14]에서의 알고리즘이 수렴하기 위한 충분조건이 주어진 시스템 및 추정된 시스템의 선형 연산자 (linear operator)들로 이루어져 있으므로, 사전에 주어진 시스템이 이 조건을 만족하는지 혹은 추정된 시스템이 어느 정도까지 주어진 시스템과 유사할 때 학습 제어 알고리즘이 수렴하는지 검토하기가 매우 어렵다. 따라서 본 논문에서는 [14]에서 제안된 방법을 사용할 때 이 방법이 수렴하기 위한 새로운 충분 조건을 제시하고 수치 예제를 통하여 타당성을 보이기로 한다. 제시된 방법이 로봇 머니플레이터의 동작제어에 유용하게 적용될 수 있음을 보이기 위하여 선형화된 모델을 이용한 로봇 반복학습 제어기를 설계하고 두링크 로봇 머니플레이터를 예제로 들어 그 유용성을 보이기로 한다.

앞으로 본 논문에서는 행렬  $B$ 의 전치(transpose) 행렬을  $B^T$ , 행렬  $B$ 의 유사역행렬(pseudo inverse matrix)을  $B^+$ , 즉  $B^+ = (B^T B)^{-1} B^T$  [9],  $n \times n$  영행렬을  $O_n$ ,  $n \times n$  단위행렬을  $I_n$ 으로 표시하기로 하고, 대각행렬을  $\text{diag}\{\cdot\}$ 로 표시하기로 한다.  $n$ 차원의 벡터  $x$ 에 대하여  $\|x\|$ 는 유클리드 노름(Euclidean norm)을 그리고,  $\|x\|_\infty$ 는 최대 노름을 뜻한다. 즉  $x = (x^1, x^2, \dots, x^n)$ 일때

$$\|x\|_\infty = \max_{1 \leq i \leq n} |x^i|$$

이다.  $n \times m$ 행렬  $G$ 에서 각각의 요소(element)를  $g^{ij}$ 라 표시하면

$$\|G\|_\infty = \max_{1 \leq i \leq n} \left\{ \sum_{j=1}^m |g^{ij}| \right\}$$

이며 또한 주어진 시간함수  $h: [O, T] \rightarrow R^n$ 에서

$$\|h(\cdot)\|_\infty = \sup_{t \in [0, T]} e^{-\lambda t} \|h(t)\|_\infty$$

으로 정의한다.

## 2. 선형 주기성 시변 시스템에 대한 반복학습 제어 방법

본 절에서는, 1절에서 설정한 제어문제  $(P)$ 를 풀기 위한 한 방법으로서 반복학습제어 방법을 제시하기로 한다. 앞으로 점차  $k$ 는 반복동작의 수를 나타내는바 예를 들어  $x_k(t)$ 는  $k$ 번째 반복동작의 시간  $t$ ,  $0 \leq t \leq T$ ,에서의 시스템의 상태벡터의 값을 뜻하고  $e_k(t)$  및  $u_k(t)$ 도 마찬가지로 정의된다.

반복학습제어 방법을 제안하기에 앞서 우선 본 방법에서 사용하는 주기성 시변 시스템의 파라미

터 추정 방법에 대하여 설명하기로 한다. 일반적으로 시변시스템의 파라미터 추정은 상당히 어려운 문제로 알려져 있는데 본 논문의 대상시스템은 시변시스템중 주기성을 갖는 시변 시스템이므로 이 성질을 이용하여 잘 알려진 시불변 시스템에 대한 순환 최소자승방법(recursive least square method)을 기존의 방법과 같이 연속 시간축  $t$ 에 대하여가 아닌 동작이 반복되는 반복 시퀀스축에 대하여 순환 공식이 유효하도록 변형하여 사용하였다. 이를 구체적으로 보면, (3)식에서 보였듯이 시스템 행렬들을 주기행렬로 가정하였으므로 매 반복시 같은 시간점에서의 시스템 행렬값은 항상 같은 값을 알 수 있다. 즉  $k=0, 1, 2, \dots$ 에 대하여  $t \in [O, T]$ 인  $t$ 에서  $A(t) = A(t+kT)$ 이고  $B(t) = B(t+kT)$ 이다. 여기서  $\tilde{A}_k(t)$  및  $\tilde{B}_k(t)$ 를  $A(t)$   $k$ -번째 동작시  $t \in [O, T]$ 에서  $A(t)$  및  $B(t)$ 의 추정된 값이라고 표현한다면  $t \in [O, T]$ 인 모든  $t$ 에 대하여  $\tilde{A}_k(t)$  및  $\tilde{B}_k(t)$ 는 과거의 추정 값인  $\tilde{A}_l(t)$ ,  $\tilde{B}_l(t)$ ,  $l=0, 1, 2, \dots, k-1$  및 입력력값을 이용하여 순환 최소자승 방법[10]을 적용하여 추정할 수 있다. 좀 더 구체적인 수식을 통하여 이를 살펴보면, (1)식의  $A(t)$  및  $B(t)$ 를 다음과 같은 형태로 다시 쓰기로 한다.

$$A(t) = \begin{bmatrix} a^1(t) \\ \vdots \\ a^n(t) \end{bmatrix}, \quad B(t) = \begin{bmatrix} b^1(t) \\ \vdots \\ b^n(t) \end{bmatrix}$$

여기서  $a^i(t)$  및  $b^i(t)$ (단  $i=1, \dots, n$ )는 각각  $1 \times m$  열벡터이다. 이를 이용하여  $1 \times (n+m)$ 벡터  $\theta^i(t)$  및  $\Psi(t)$ 를 다음과 같이 정의한다.

$$\theta^{iT}(t) = [a^i(t) b^i(t)], \quad i=1, 2, \dots, n \quad (5)$$

$$\Psi(t) = [x^T(t) u^T(t)]^T$$

이제 모든  $t \in [O, T]$ 에 대하여  $A(t)$ 의  $n^2$  파라미터 및  $B(t)$ 의  $nm$  파라미터를 추정하기로 하는바, 이를 위하여

$$x^T(t) = [x^1(t), x^2(t), \dots, x^n(t)] \\ y(t) = \dot{x}(t) \quad (6)$$

와 같이  $y(t)$ 을 정의하자. 여기서  $n \times 1$ 벡터  $y(t)$ 의  $i$ 번째 항  $y^i(t)$ 은 물론  $\dot{x}(t)$ 이다.

$$y^i(t) = \theta^{iT}(t) \Psi(t), \quad i=1, 2, 3, \dots, n \quad (7)$$

이제 (7)식의 관계로부터 잘 알려진 순환 최소자승 파라미터 추정 알고리즘을  $t$ 를 고정시키고  $k$ 에 대하여 적용할 수 있다. 즉  $t \in [O, T]$ 인 모든  $t$

에 대하여  $k$ 번째 반복동작에서의 추정된 파라미터 벡터  $\hat{\theta}_k^i(\tau)$ 은 다음 식으로 구해진다.

$$\hat{\theta}_k^i(\tau) = \hat{\theta}_{k-1}^i(\tau) + F_k(\tau) [y_k^i(\tau) - \hat{\theta}_{k-1}^i(\tau) \Psi_k(\tau)] \quad (8)$$

$$F_k^i(\tau) = \frac{S_{k-1}^i(\tau) \Psi_k^T(\tau)}{\frac{1}{\alpha_k^i(\tau)} + \Psi_k^T(\tau) S_{k-1}^i(\tau) \Psi_k(\tau)} \quad (9)$$

$$S_k^i(\tau) = S_{k-1}^i(\tau) - \frac{S_{k-1}^i(\tau) \Psi_k(\tau) \Psi_k^T(\tau) S_{k-1}^i(\tau)}{\frac{1}{\alpha_k^i(\tau)} + \Psi_k^T(\tau) S_{k-1}^i(\tau) \Psi_k(\tau)} \quad (10)$$

여기서 초기 파라미터  $\hat{\theta}_0^i(\tau)$  및 초기 가중행렬  $S_0^i(\tau)$ ,  $\tau \in [0, T]$ ,  $i=1, 2, \dots, n$ 은 기존의 순환 최소자승 추정알고리즘[10]에서와 같이 적절히 주어지는 값이고,  $\alpha_k^i(\tau)$ 은 상실인자(forgetting factor)로써  $0 < \alpha_k^i(\tau) \leq 1$ 인 범위내에서 적당한 값으로 선택한다. 여기서 만일 측정 데이터  $y(t)$ (식(8))가 잡음(noise) 성분을 많이 포함하고 있으면 앞에서 제시한 추정 방법은 일반적으로 사용하기가 어려움게 되고 이런 경우에는 필터등의 적절한 방법이 요구된다. 그러나 로보트 머니플레이터의 경우에는  $y(t)$ 는 타코메타(tachometer)나 가속도 측정장치(accelerometer)등을 이용하여 잡음성분이 없는  $y(t)$ 를 측정할 수 있어 앞에서 제시한 알고리즘을 유용하게 사용할 수 있다.

이제 추정된 시스템 행렬  $\hat{A}_k(t)$  및  $\hat{B}_k(t)$ 를 이용하여 반복학습제어를 제시하기로 한다. 우선

$$e_k(t) = x^D(t) - x_k(t), \quad 0 \leq t \leq T, \quad k=0, 1, 2, 3, \dots \quad (11)$$

와 같이 오차벡터를 정의 하고  $t \in [0, T]$ 인 모든  $t$ 에서 또한  $k=0, 1, 2, 3, \dots$ 인 모든  $k$ 에 대하여  $n \times m$ 행렬  $\hat{B}_k(t)$ 의 랭크가  $m$ 이라고 가정하자. 그러면  $\hat{B}_k(t)$ 의 유사 역행렬  $\hat{B}_k^+(t)$ 가 항상 존재하고 이를 이용하여 다음과 같이  $\hat{u}_k(t)$ 를 정의한다.

$$\hat{u}_k(t) = \hat{B}_k^+(t) [\dot{e}_k(t) - \hat{A}_k(t) e_k(t)], \quad 0 \leq t \leq T \quad (12)$$

이 (12)식은 (1)식으로 주어진 시스템의 역 시스템 모델(inverse system model)이라고 하는 데 즉 (1)식은  $u_k(t)$ 의 입력이 들어가  $A(t)$  및  $B(t)$ 를 통하여  $x(t)$ 가 만들어지는 반면 (12)식은  $e_k(t)$  및  $\dot{e}_k(t)$ 가 입력되어  $\hat{A}_k(t)$  및  $\hat{B}_k(t)$ 를 통하여  $\hat{u}_k(t)$ 를 만들어 내는 시스템을 말한다. 또한  $\Psi(t)$ 를 다음 식으로 주어지는 기본 행렬 차등식(fundamental matrix differential equation)의 유일해(unique solution)이라고 하자.

$$\dot{\Psi}(t) = A(t) \Psi(t), \quad \Psi(0) = I_n \quad (13)$$

이제 반복학습제어 알고리즘을 다음과 같이 제안한다.

(반복 학습제어 알고리즘)

상태 제적 오차한계  $\epsilon^* > 0$ 이 주어지고 초기 시행시 입력  $u_0(t)$ ,  $0 \leq t \leq T$ 가  $u_0(\cdot) \in C^n[0, T]$ 를 만족하도록 주어졌다고 하자. 또한 초기 추정 시스템 행렬  $\hat{A}_0(t)$ 와  $\hat{B}_0(t)$ ,  $0 \leq t \leq T$ 가  $[0, T]$ 에서 연속이며  $\hat{B}_0(t)$ 의 랭크가  $m$ 이 되도록 주어졌다고 하자. 그리고  $k=0$ 으로 한다.

단계1: 시스템 상태벡터를 측정하여 오차를 계산한다. 즉

$$e_k(t) = x^D(t) - x_k(t) \quad (14)$$

이고

$$x_k(t) = \Psi(t) x_k(0) + \int_0^t \Psi(t-\tau) B(\tau) u_k(\tau) d\tau \quad (15)$$

이다.

단계2: 역시스템 모델을 이용하여  $\hat{u}_k(t)$ 를 계산한다. 즉

$$\hat{u}_k(t) = \hat{B}_k^+(t) [\dot{e}_k(t) - \hat{A}_k(t) e_k(t)] \quad (16)$$

단계3: 다음 시행 입력을 다음과 같이 계산한다.

$$u_{k+1}(t) = u_k(t) + \hat{u}_k(t) \quad (17)$$

단계4: 만일  $\|e_k(t)\|_\infty \leq \epsilon^*$ 이면 정지하고 아니면  $k=k+1$ 로 하여 단계1로 간다.

상기의 반복학습 제어기의 구조를 그림1에 보였으며 제안한 반복 학습제어 알고리즘의 수렴성을 다음의 정리1에 요약하였다.

**정리 1**

(1)식으로 주어진 선형 주기성 시변 시스템에 대하여, (8)~(10)식으로 주어진 파라미터 추정 알고리즘이 아래의 부등식 (18)식을 만족하고,

$$\|I - \hat{B}_k^+(t) B(t)\|_\infty < 1 \quad (18)$$

모든  $k$  및 모든  $t \in [0, T]$ 에 대하여  $n \times m$ 행렬  $\hat{B}_k(t)$ 의 랭크가  $m$ 이며 매 반복시행시  $x_k(0) = x^D(0) = \xi^0$ 이면 (14)~(17)식으로 제안한 반복 학습제어기에 의하여 시스템의 상태 벡터는  $k$ 가 증가함에 따라 원하는 제적을 오차없이 추종한다. 즉  $\lim_{k \rightarrow \infty} \|e_k\|_\infty = 0$ 이다.

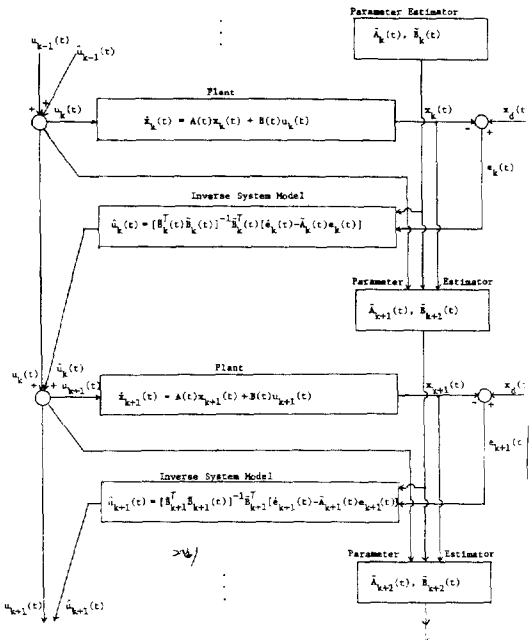


그림 1 제안한 학습제어 시스템의 블록선도  
 Fig. 1 A Schematic Diagram of the Proposed Learning Control System

(증명) :

$u^D(t)$ 를  $x(t)$ 로 하여금 원하는 궤적  $x^D(t)$ 를 쫓아가도록 하게 하는 제어입력이라고 하자. 이때 (1), (14), (16), (17)식 들로부터

$$\begin{aligned}
 & u^D(t) - u_{k-1}(t) \\
 &= u^D(t) - \tilde{u}_k(t) - u_k(t) \\
 &= u^D(t) - u_k(t) - \tilde{B}_k^+(t) [\dot{e}_k(t) - \tilde{A}_k(t)e_k(t)] \\
 &= u^D(t) - u_k(t) - \tilde{B}_k^+(t) [\dot{x}^D(t) - \dot{x}(t) \\
 &\quad - \tilde{A}_k(t)\{x^D(t) - x_k(t)\}] \\
 &= u^D(t) - u_k(t) - \tilde{B}_k^+(t) \{ [A(t) - \tilde{A}_k(t)] \\
 &\quad \{x^D(t) - x_k(t)\} + B(t)\{u^D(t) - u_k(t)\} \} \\
 &= \{I - \tilde{B}_k^+(t)B(t)\} \{u^D(t) - u_k(t)\} \\
 &\quad - \tilde{B}_k^+(t) \{A(t) - \tilde{A}_k(t)\} \{x^D(t) - x_k(t)\} \quad (19)
 \end{aligned}$$

가 얻어진다. 좌우 양변의 노름을 계산하면 모든  $t \in [0, T]$ 에 대해 다음과같은 부등식이 얻어진다.  
 $\|u^D(t) - u_{k+1}(t)\|_\infty \leq \|I - \tilde{B}_k^+(t)B(t)\|_\infty \|u^D(t) - u_k(t)\|_\infty + \|\tilde{B}_k^+(t) \{A(t) - \tilde{A}_k(t)\}\|_\infty \|x^D(t) - x_k(t)\|_\infty$  (20)

가정에서 모든  $k$ 에 대해  $x_k(0) = x^D(0)$ 이 성립하므로 모든  $t \in [0, T]$ 에서 다음의 관계가 성립한다.

$$\|x^D(t) - x_k(t)\|_\infty = \left\| \int_0^t \{ [A(\tau)x^D(\tau) + B(\tau)u^D(\tau)] \right.$$

$$\begin{aligned}
 & \left. - [A(\tau)x_k(\tau) + B(\tau)u_k(\tau)] \} d\tau \right\|_\infty \\
 & \leq \int_0^t \{ \| [A(\tau)x^D(\tau) - x_k(\tau)] \| \\
 & \quad + \| B(\tau)\|_\infty \| u^D(\tau) - u_k(\tau) \| \} d\tau \quad (21)
 \end{aligned}$$

(21)식에서 Bellman-Gronwall의 보조정리(Lemma) [12]를 적용하면, 모든  $t \in [0, T]$ 에 대해 (22)식이 성립한다.

$$\|x^D(t) - x_k(t)\|_\infty \leq \int_0^t \|B(\tau)\|_\infty \|u^D(\tau) - u_k(\tau)\|_\infty e^{a(t-\tau)} d\tau \quad (22)$$

여기서 모든  $t \in [0, T]$ 에 대해  $a = \|A(t)\|_\infty$ 이다.

따라서 (21)식과 (22)식으로부터 (23)식이 얻어진다. 이후부터 식을 간단히 하기 위해서  $\delta u_k(t) = u^D(t) - u_k(t)$ 로 표기한다.

$$\begin{aligned}
 & \|\delta u_{k+1}(t)\|_\infty \leq \|I - \tilde{B}_k^+(t)B(t)\|_\infty \|\delta u_k(t)\|_\infty \\
 & \quad + \|\tilde{B}_k^+(t) \{A(t) - \tilde{A}_k(t)\}\|_\infty \\
 & \quad \int_0^t \|B(\tau)\|_\infty \|\delta u_k(\tau)\|_\infty e^{a(t-\tau)} d\tau \quad (23)
 \end{aligned}$$

(23)식의 양변에 정함수(positive function)  $e^{-\lambda t}$ 를 곱하면, 모든  $t \in [0, T]$ 에 대해 다음의 관계가 얻어진다.

$$\begin{aligned}
 & e^{-\lambda t} \|\delta u_{k+1}(t)\|_\infty \leq \|I - \tilde{B}_k^+(t)B(t)\|_\infty e^{-\lambda t} \|\delta u_k(t)\|_\infty \\
 & \quad + \|\tilde{B}_k^+(t) \{A(t) - \tilde{A}_k(t)\}\|_\infty e^{-\lambda t} \int_0^t \|B(\tau)\|_\infty \|\delta u_k \\
 & \quad (\tau)\|_\infty e^{a(t-\tau)} d\tau \leq \|I - \tilde{B}_k^+(t)B(t)\|_\infty e^{-\lambda t} \|\delta u_k \\
 & \quad (t)\|_\infty + \|\tilde{B}_k^+(t) \{A(t) - \tilde{A}_k(t)\}\|_\infty b \int_0^t e^{-\lambda \tau} \|\delta u_k \\
 & \quad (\tau)\|_\infty e^{(a-\lambda)(t-\tau)} d\tau \quad (24)
 \end{aligned}$$

여기서 모든  $t \in [0, T]$ 에 대해  $b = \|B(t)\|_\infty$ 이다. 따라서 모든  $t \in [0, T]$ 에 대해, 그리고  $\lambda \neq a$ 인  $\lambda$ 에 대해

$$\begin{aligned}
 & \|\delta u_{k+1}(\cdot)\|_\lambda \leq \|I - \tilde{B}_k^+(t)B(t)\|_\infty \|\delta u_k(\cdot)\|_\lambda \\
 & \quad b \|\tilde{B}_k^+(t) \{A(t) - \tilde{A}_k(t)\}\|_\infty \\
 & \quad \left[ \frac{1 - e^{-(a-\lambda)T}}{\lambda - a} \right] \|\delta u_k(\cdot)\|_\lambda \quad (25)
 \end{aligned}$$

가정으로 부터  $\|I - \tilde{B}_k^+(t)B(t)\|_\infty < 1$ 이므로  $\lambda > 0$ 를 충분히 크게 선택함으로써 다음의 부등식이 성립하도록 할 수 있다.

$$\|I - \tilde{B}_k^+(t)B(t)\|_\infty + b \|\tilde{B}_k^+(t) \{A(t)\}$$

$$-\tilde{A}_k(t)\|_{\infty} \left[ \frac{1 - e^{(a-\lambda)T}}{\lambda - a} \right] < 1 \tag{26}$$

따라서  $k$ 가 커짐에 따라  $\|\delta u_k(\cdot)\|_{\infty}$ 는 0으로 수렴한다. 한편  $\|\cdot\|_{\infty}$ 의 정의로부터

$$\sup_{t \in [0, T]} \|\delta u_k(t)\|_{\infty} \leq e^{\lambda T} \|\delta u_k(\cdot)\|_{\infty}$$

이다. 결국  $k$ 가 커짐에 따라  $\sup_{t \in [0, T]} \|\delta u_k(t)\|_{\infty}$ 가 0으로 수렴하게 된다. 이는 곧 모든  $t \in [0, T]$ 에 대해  $k \rightarrow \infty$ 일 때  $u_k(t) \rightarrow u^D(t)$ 를 의미하며 (21)식에 의해서 모든  $t \in [0, T]$ 에 대해  $k \rightarrow \infty$ 일 때  $x_k(t) \rightarrow x^D(t)$ 가 성립함을 알 수 있다. (증명끝)

**주 1**

위의 제어 알고리즘에서  $y(t) = I \cdot x(t)$ 라고 두면, 입력 벡터와 출력 벡터와의 차수(order)차는 1이다. 그러면  $\dot{y}(t) = \dot{x}(t) = A(t)x(t) + B(t)u(t)$ 이다. 따라서  $B(t)$ 는 입력에서 출력의 미분값으로의 직달항(direct transmission term)이다. 이러한 고찰과 정리1의 충분조건에서 보면 Sugie등[13]이 관찰했던 직달항이 수렴성 증명에 결정적인 역할을 한다는 사실과 일치한다.

**주 2**

정리1의 충분조건에서 보면 추정된 파라미터가 실제값으로 가지 않아도 수렴함을 알 수 있다. 즉  $\tilde{A}_k(t)$ 는 임의의 제한된(bound) 값으로 수렴하고,  $\tilde{B}_k(t)$ 는 (21)식을 만족하면 알고리즘의 수렴성이 보장됨을 알 수 있다. [14]에서는 충분조건이 선형연산자로 서술되었으므로 추정된 모든 파라미터가 수렴성에 관련되어 있으나, 정리1의 결과로 보면 행렬  $B(t)$ 에만 관련이 있으므로 조건을 검토하기가 쉬울 뿐만 아니라 적용 범위가 넓어졌음을 알 수 있다.

**주 3**

Arimoto등의 알고리즘과 제안된 알고리즘을 비교해보면,  $\tilde{B}_k^*(t)$ 는  $\Gamma(t)$ 와 대응되며,  $\tilde{B}_k^*(t)\tilde{A}_k(t)$ 는 Arimoto등의 알고리즘에서는 0이다. 식(19)에서 만약  $A(t)$ 와  $\tilde{A}_k(t)$ 가 같으면 우변의 2번째 항은 0으로 되지만 Arimoto등의 알고리즘에서는 항상 상수 값으로 남는다. 따라서 파라미터 추정치가 실제값에 가까우면 다음 절의 시뮬레이션 결과에서 보듯이 수렴 속도가 훨씬 빠르게 됨을 알 수 있다.

**3. 수치 예제들**

제안한 학습 제어 알고리즘의 타당성을 보이기 위하여 단일 입출력 선형 시변 시스템 및 두개의 링크를 갖는 로봇트 머니플레이터에 대하여 각각 컴퓨터 시뮬레이션을 수행하였다.

(예제1): 선형 시변 시스템

단일 입출력 선형 시변 시스템이 다음과 같이 주어졌다고 하자.

$$\begin{aligned} \begin{bmatrix} \dot{x}^1(t) \\ \dot{x}^2(t) \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -(2+5t) & -(3+2t) \end{bmatrix} \begin{bmatrix} x^1(t) \\ x^2(t) \end{bmatrix} \\ &+ \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), \begin{bmatrix} x^1(0) \\ x^2(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, 0 \leq t \leq 1 \end{aligned} \tag{27}$$

또한 원하는 상태 궤적 및 허용 오차한계를 다음과 같이 선정하였다.

$$\begin{aligned} x^{1D}(t) &= t^3(4-3t) \\ x^{2D}(t) &= 12t^2(1-t) \\ \epsilon &= 0.005 \end{aligned} \tag{28}$$

앞 절에서 제안한 학습 제어방법을 적용하기 위하여 초기 입력  $u_0(t)$  및 초기 추정 시스템 행렬  $\tilde{A}_0(t)$  및  $\tilde{B}_0(t)$ 를 다음과 같이 선택하였다.

$$\begin{aligned} u_0(t) &= 0, 0 \leq t \leq 1 \\ \tilde{A}_0(t) &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \\ \tilde{B}_0(t) &= \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}, 0 \leq t \leq 1 \end{aligned} \tag{29}$$

또한 파라미터 추정기의 초기 조건을 다음과 같이 선택하였다.

$$\begin{aligned} \alpha_k^i(t) &= 1, i=1, 2, k=0, 1, 2, \dots, 0 \leq t \leq 1 \\ S_0^i(t) &= \text{diag}\{10000, 10000, 10000\}, 0 \leq t \leq 1 \end{aligned}$$

이상의 데이터를 이용하여 제안한 반복 학습 제어 알고리즘을 적용한 결과 각 상태변수의 궤적을 그림2에 보였다. 그림2에서 보면, (27)식의 시스템에 비하여 (29)식과 같이 모델링 오차가 상당히 큰 경우에도 5번의 반복 동작 후에는 시스템의 상태벡터 궤적이 원하는 상태궤적을 허용오차한계 내에서 잘 쫓아가고 있음을 알 수 있다.

본 논문에서 제안한 알고리즘과 기존의 학습 제어 알고리즘을 비교하기 위하여 [6]에서 제안한 알고리즘을 전술한 예제에 적용하여 보았다. 이를

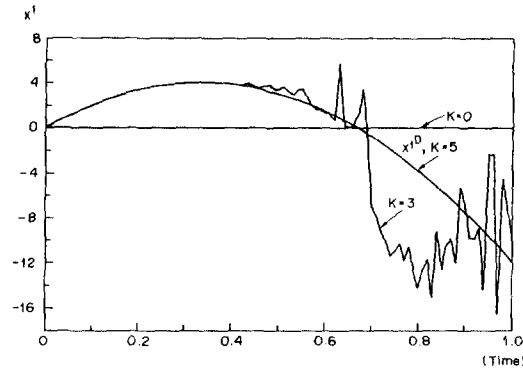
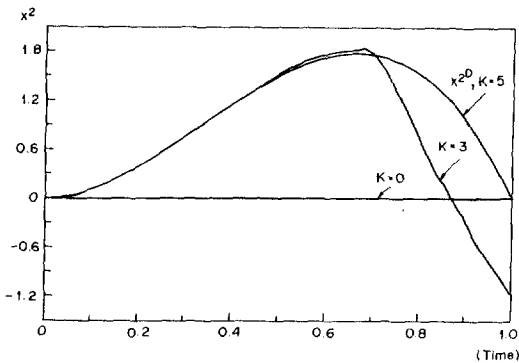


Fig. 2 State Trajectories by the Proposed Learning Algorithm

(a)  $x^1(t)$  Trajectories



(b)  $x^2(t)$  Trajectories

그림 2 제안한 학습 알고리즘에 의한 상태 궤적  
(a)  $x^1(t)$  궤적  
(b)  $x^2(t)$  궤적

Fig. 2 State Trajectories by the Proposed Learning Algorithm  
(a)  $x^1(t)$  Trajectories  
(b)  $x^2(t)$  Trajectories

위하여 Arimoto의 알고리즘을 요약하면 다음과 같다.

$$y_k = C(t) \Psi(t) x_k(0) + \int_0^t C(t) \Psi(t) \Psi^{-1}(\tau) B(\tau) u_k(\tau) d\tau \quad (30)$$

$$e_k(t) = y^D(t) - y_k(t) \quad (31)$$

$$u_{k+1}(t) = u_k(t) + \Gamma(t) \dot{e}_k(t) \quad (32)$$

여기서  $y$ 는  $r$ 차원의 출력벡터이고  $C$ 는  $rxn$  출력 행렬이다. (25)~(27)식으로 제안된 알고리즘의 수렴조건은 [6]

$$\|I_r - C(t)B(t)\Gamma(t)\|_\infty < 1 \quad (33)$$

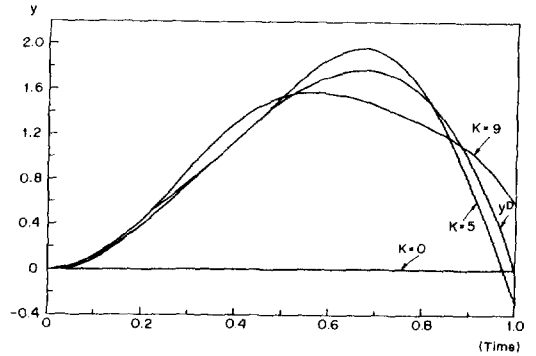


그림 3 Arimoto 알고리즘[6]에 의한 출력 궤적  
Fig. 3 Output Trajectories by the Arimoto's Algorithm[6]

로 주어지는데 본 예제의 경우  $C = [0 \ 1]$ 로 하였고  $B(t)$ 는 (29)식에서 주어진 값으로 하였다. 이와 같은 경우 가중인자  $\Gamma(t)$ 는 스칼라 상수  $\gamma$ 로 되고 수렴속도를 최대로 하기 위해 (36)식을 만족하는  $\gamma$ 중 최소값으로 하였다. 이때  $\gamma=2$ 로 되고  $\gamma=2$ 로 하여 (30)~(32)식의 알고리즘을 본 예제에 적용한 결과 출력 궤적을 그림3에 보였다. 그림3에서 보듯이 10번의 반복 시행을 행한 후에도 원하는 궤적과 출력 궤적 사이의 절대값 오차는 0.21만큼 남아있는 것을 알 수 있다. 또한 만일 시스템 다이내믹스를 완전하게 알고 있다고 가정하는 경우  $\gamma$ 값은 1로 되는데  $\gamma=1$ 로 하여 Arimoto의 알고리즘을 적용한 결과 최대 절대값 오차는 10번의 반복 시행 후에도 0.06만큼 남아 있음을 알 수 있었다, 이상으로부터 [6]에서 제안한 알고리즘이 본 논문에서 제안한 알고리즘보다 제어기의 형태는 간단하지만 수렴속도 면에서는 본 알고리즘이 훨씬 우수하다는 것을 알 수 있다.

<예제2>: 로봇트 머니플레이터의 연속 경로 제어에의 응용

앞 절에서 제안한 반복 학습제어 알고리즘을  $n$  관절 머니플레이터에 응용하기 위하여 원하는 명목 궤적(nominal trajectory)이 주어졌고 주어진 명목 궤적을 쫓아가기 위한 명목 입력 토오크를 로봇트 다이내믹스의 Newton-Euler식을 이용하여 계산하였다고 가정하자. 그러나 명목 입력 토오크를 계산할 때 사용하는 로봇트 다이내믹스의 모델식은 마찰이나 백래쉬(backlash) 및 비탄성 특성(nonrigidity) 등의 모델링하기 어려운 항목들로 인하여 모델식 자체가 상당한 오차를 갖을 수 있고 따라서 모델링 오차가 존재하는 모델로부터 계산

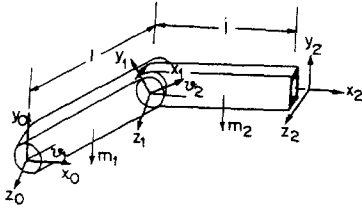


그림 4 2-링크 로봇트 아암  
Fig. 4 A Two-Link Robot Arm

된 명목 입력 토오크를 로봇트 머니플레이터에 가한 경우 머니플레이터의 동작 계획은 주어진 원하는 명목 계획의 근점 주변을 쫓아갈 뿐 정확한 계획 추적을 얻기는 상당히 어렵게 된다. 이렇게 여전히 남아있는 계획 오차를 없애기 위하여 부가적인 토오크 입력을 제안한 반복 학습제어기에 의하여 제공하고자 한다. 이를 좀더 구체적으로 보기 위하여, 우선 로봇트 머니플레이터의 비선형 다이나믹스 식을 원하는 명목계획 주변에서 선형화 시키면 잘 알려진 바와 같이 선형 시변 시스템으로 표현할 수 있다. 따라서 2절에서 제안한 바 (8) - (10)식의 파라미터 추정기 및 (14) - (17)식의 반복 학습제어 알고리즘을 선형화된 로봇트 머니플레이터의 경로제어에 직접 응용할 수 있다.

로봇트 머니플레이터의 연속 경로제어에 제안한 반복 학습제어 알고리즘의 응용 가능성 및 유효성을 보이기 위하여 그림4에 보인 바와 같이 회전관절을 갖는 2링크 로봇트 머니플레이터에 대하여 컴퓨터 시뮬레이션을 행하였다. 로봇트 머니플레이터의 동작 식은 다음과 같이 주어진다[11]. :

$$\begin{bmatrix} \tau^1 \\ \tau^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{3}m_1l^2 + \frac{4}{3}m_2l^2 + m_2C_2l^2 \frac{1}{3}m_2l^2 + \frac{1}{2}m_2C_2l^2 \\ \frac{1}{3}m_2l^2 + \frac{1}{2}m_2C_2l^2 \end{bmatrix} + \begin{bmatrix} \dot{\theta}^1 \\ \dot{\theta}^2 \end{bmatrix} + \begin{bmatrix} -\frac{1}{2}m_2S_2\dot{\theta}^{22} - m_2S_2 + l^2\dot{\theta}^1\dot{\theta}^2 \\ \frac{1}{2}m_2S_2l^2\dot{\theta}^{12} \end{bmatrix} + \begin{bmatrix} -\frac{1}{2}m_1glC_1 - \frac{1}{2}m_1glC_{12} - m_2glC_1 \\ -\frac{1}{2}m_2glC_{12} \end{bmatrix} \quad (34)$$

여기서  $C_i$ ,  $S_i$  및  $C_{ij}$ 는 각각  $\cos\theta^i$ ,  $\sin\theta^i$  및  $\cos(\theta^i + \theta^j)$ 를 나타내고  $\theta^1$  및  $\theta^2$ 는 관절변수이다. 또한 시뮬레이션에서 사용한 링크 질량 및 길이를

표 1 시뮬레이션에서 사용한 데이터

Table 1 Input Data for Simulation

Parameter	Value	True Value	Modelled Value
mass of link 1		2 kg	1.8 kg
mass of link 2		2 kg	2.2 kg
length		0.5 m	0.45 m

표1에 보였다. (37)식을 Lagrange-Euler식의 표준화된 형태로 쓰면 다음과 같이 쓸수 있다.

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

원하는 계획 및 허용 오차한계  $\epsilon$ 를 다음과 같이 선정하였다.

$$\theta^{1D} = -\frac{5}{3}t^3 + \frac{5}{2}t^2 \quad (35)$$

$$\theta^{2D} = \frac{5}{3}t^3 - \frac{5}{2}t^2 - \frac{5}{6}$$

$$\epsilon = 1.0[^\circ] = 0.017[\text{radian}] \quad (36)$$

(14) - (17)식의 제안된 알고리즘을 적용하기 위하여 우선 Newton-Euler식을 이용하여 명목 토오크 입력  $\tau^N(t)$ 를 구해야 하는데, 전술한 바 모델링 오차를 고려하여 표1에 보인 바와 같이 실제 파라미터 값과 모델링된 파라미터 값을 다르게 하여  $\tau^N(t)$ 를 구하였다. 또한 초기 시행시 입력 및 초기 추정 시스템 행렬은 다음과 같이 선정하였다.

$$u_0(t) = 0, \quad 0 \leq t \leq 1 \quad (37)$$

$$\tilde{A}_0(t) = \begin{bmatrix} O_2 & \cdot & I_2 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ & & O_{2 \times 4} \end{bmatrix},$$

$$\tilde{B}_0(t) = \tilde{D}^{-1}(\theta^D(t)), \quad 0 \leq t \leq 1$$

(37)식에서  $\tilde{D}(\theta)$ 는 관성행렬  $D(\theta)$ 에 대한 모델로써 불확실한 파라미터를 갖고 있음을 뜻한다. 끝으로 (8) - (10)식으로 주어진 파라미터 추정기에 대한 초기조건을 다음과 같이 선정하였다.

$$\alpha^i_k(t) = 0.93, \quad i = 1, 2, \quad k = 0, 1, 2, \dots, 0 \leq t \leq 1 \quad (38)$$

$$S^i_0(t) = \text{diag}\{10^5, \dots, 10^5\}, \quad i = 1, 2, \quad 0 \leq t \leq 1$$

이상의 데이터로부터 시뮬레이션을 행한 결과 관절각 계획을 그림 5에 보였다. 그림5에서 보듯이 표1에 보인 바와 같이 파라미터 불확실성이 상당히 큰 경우에도 20번의 반복시행 후에도 관절각 계획이 원하는 계획을 허용오차 한계 내에서 잘



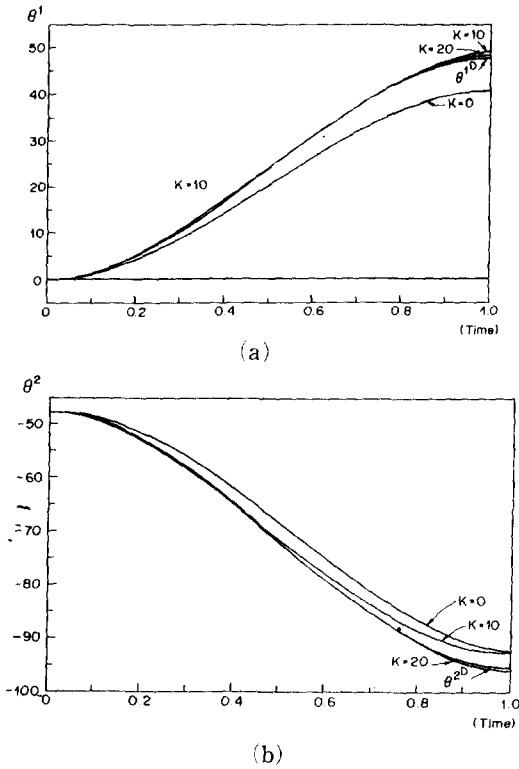


그림 5 제안한 학습 알고리즘에 의한 관절각 궤적  
 - (a) 관절각 1  
 (b) 관절각 2

Fig. 5 Joint Angle Trajectories by the Proposed Learning Algorithm  
 (a) Joint Angle 1  
 (b) Joint Angle 2

찾아가고 있음을 알 수 있다.

#### 4. 결론

선형 주기성 시변 시스템에 대한 새로운 반복 학습제어 알고리즘을 제안하였는 바 제안된 알고리즘에서는 주기성 시변 시스템의 파라미터 추정을 효과적으로 하기 위하여 동작 반복축에 대하여 순환 공식이 적용되는 파라미터 추정기를 사용하였다. 제안된 알고리즘이 수렴하기 위한 충분조건을 제시하였고 충분조건에서 시스템의 행렬  $A(t)$ 에는 관련이 없고, 입력 행렬  $B(t)$ 에 관련이 있음을 알 수 있었다. 또한 제안된 알고리즘이 유용하게 적용될 수 있음을 보였다. 그러나 제안된 알고

리즘을 로봇트 동작제어에 이용할 경우 명목 궤적에 대하여 외란이 큰 경우에는 선형화된 시스템의 선형성이 상실되므로 이러한 경우에는 제안된 알고리즘을 그대로 적용할 수 없는 단점을 갖고 있다. 따라서 큰 외란이나 파라미터 불확실성이 심한 경우에도 일반적으로 적용할 수 있는 강인한 로봇트 반복 학습 제어기의 설계에 대한 연구는 앞으로의 연구과제로 남아있다.

#### 참고 문헌

- [1] H.A. Simon, "Why Should Machine Learn?" Machine Learning, Springer-Verlag, Berlin Heidelberg, New York, 1984.
- [2] J.G. Carbonnel, R.S. Michalski and T.M. Mitchell, "An Overview of Machine Learning," *ibid* pp. 3-23.
- [3] 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.
- [4] A.J. Koivo and T-H. Guo, "Adaptive Linear Controller for Robotic Manipulator," *IEEE Trans. Automat. Contr.*, vol. AC-28, no. 2, 1983.
- [5] M.R. Raibert, "Motor Control and Learning by the State Space Model," Ph. D Dissertation, Dept. of Psychology, MIT, 1977.
- [6] S. Arimito, S. Kawamura and F. Miyazaki, "Bettering Operation of Robots by Learning," *Journal of Robotic Systems*, vol. 1, no. 2, 1984.
- [7] S. Arimito, S. Kawamura and F. Miyazaki, "Can Mechanical Robots Learn by Themselves?" *Proc. of 2nd Int. Symp. Robotics Res.* Kyoto, Japan, August 1984.
- [8] S. Kawamura, F. Miyazaki and S. Arimito, "Iterative Learning Control for Robotic System," *Proc. of IECON 84*, Tokyo, Japan, 1984.
- [9] S. Barnett, *Matrices in Control Theory*, New York, NY: Van Nostrand Reinhold, 1971.
- [10] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*, MIT Press, London, England, 1983.
- [11] C.S.G. Lee, R.C. Gonzalez and K.S. Fu, *Tutorial on Robotics*, IEEE Computer Society

- Press, 1984.
- [12] V. Lakshmikantham and S. Leela, *Differential and Integral Inequalities*, New York, NY: Academic Press, 1969, vol. 1, pp. 37-41.
- [13] T. Sugie and T. Ono, "On a learning control law," *System and Control*. vol. 31, no. 2, pp. 129-135, 1987.
- [14] S-R. Oh, Z. Bien and I.H. Suh, "An Iterative Learning Control Method with Application for Robot Manipulator," *IEEE J. Robotics Automat.*, vol 4, no. 5, pp. 508-514, 1988.
-