

선형 이산 시스템의 학습제어 알고리즘

박희재<sup>o</sup>, 조형석  
한국과학기술원 생산공학과

현봉섭  
한국과학기술원 기계공학부

A learning control algorithm for the linear discrete system.

H.J. Park<sup>o</sup>, H.S. Cho  
Department of Production Engineering  
Korea Advanced Institute of Science and Technology

B.S. Hyun  
Division of Mechanical engineering

ABSTRACT

In this paper, an iterative leaning control algorithm for the linear discrete system is proposed. Based upon the parameter estimation method, the learning for good tracking control is acured through a sequence of repetitive operations. A series of simulation studies are performed to show the validity of this algorithm.

1. 서론

본 논문의 목적은 미지의 선형 이산 시스템의 반복 학습 제어를 설계하는 것이다. 일반적으로, 학습제어는 과거의 작업에서 축적된 지식을 바탕으로 다음 작업의 성능을 점진적으로 개선하는 제어 기법으로서, 같은 작업을 반복적으로 수행하는 공정 즉 자동조립, 프레스, 스폿용접 등에 적용이 가능하다.

최근 수 년전부터, 인간만이 가지고 있는 학습능력을 공학 시스템에 부여하고자하는 많은 연구가 진행되었다. 특히, 자동제어 분야에서는 Arimoto[1], Fruta[2], 오 상록[3] 등의 연구가 두드러진다. 이들 연구는 크게 Arimoto 와 Furuta 의 오차 기준 학습제어 ( Error based learning

control ) 와 오 상록의 모델 기준 학습제어 ( Model based learning control ) 로 분류된다. 입력 기준 학습제어는 오차 또는 오차의 미분치를 근거로 시스템을 안정화하는 제어입력을 발생시키는 방법이다. 그 구조가 매우 간단한 반면에 시스템에 대한 사전지식이 필요하며, 특정 작업에대한 학습이 끝난 후에도 작업이 바뀌면 또 다시 학습을 해야 한다. 반면에 모델 기준 학습제어 방법은 매개변수 추정을 도입하여 대상 시스템의 모델을 식별 ( Identification ) 하므로, 미지의 시스템에 적용이 가능하고, 작업이 바뀌어도 재학습의 필요가 없는 것이 장점이다. 이러한 연구 의 한계는 모두가 연속시스템에 대한 연구들 이었고, 기존 입력과 제어대상의 초기치가 일치한다는 가정하에서 안정성 증명을 하였다.

본 연구에서 제안한 학습제어 이론은 모델 기준 학습제어 방법으로서 매개변수 식별 ( Parameter estimation ) 을 통하여 미지의 대상 시스템의 매개변수를 찾아내고, 이 식별된 매개변수와 과거의 입출력 자료로 부터 다음 작업에 사용될 새로운 제어입력을 찾아낸다. 대상 시스템을 선형 이산 시스템으로 가정하여 초기 오차가 존재하는 경우 에도 그 안정성과 수렴성을 증명 하였고, 전산기 시뮬레이션을 통하여 그 효용성을 고찰하였다.

## 2. 반복 학습 제어

### 1) 대상 시스템

제어 대상 시스템은 안정한 선형 이산 시스템으로 가정한다.

$$\begin{aligned} X_{(k+1)} &= \Phi X_{(k)} + \Gamma U_{(k)}, X_{(0)} = X_0 \\ Y_{(k)} &= C X_{(k)} \end{aligned} \quad (1)$$

여기서  $\Phi$ :  $n \times n$  system matrix.  
 $\Gamma$ :  $n \times 1$  input vector.  
 $C$ :  $1 \times n$  output vector

### 2) 문제의 정의

반복학습제어 문제를 정의하면 과거의 작업에서 얻어진 지식  $\Xi_i$ 로부터 미지의 시스템 (1) 이 다음 작업에서 원하는 경로를 잘 추종할 수 있는 제어 입력  $U_{i+1}$ 를 구하는 알고리즘을 찾는 것이다.

즉, 모든 상태변수  $X_{(k)}$  가

$$\|X_{d(k)} - X_{(k)}\|_n \leq \varepsilon^* \quad , t \in [0, T] \quad (2)$$

라는 기준을 유한한 작업 횟수 안에 만족 하도록 새로운 입력

$$U_{i+1(k)} = L(\Xi_i, t) \quad , t \in [0, T] \quad (3)$$

를 찾는 것이다.

여기서  $\|\cdot\|_n$ :  $n$ -dimensional maximum-norm.  
 $X_{d(k)}$ : desired state  
 $\varepsilon^*$ : tolerance bound of control.  
 $t = k \cdot h$   
 $h$ : step size  
 $T$ : final time of each operation

$L(\cdot)$ : a function

$\Xi_i$ : previous data set

$$\begin{aligned} &= \{ X_{0(k)}, \dots, X_{i(k)}, U_{0(k)}, \\ &\dots, U_{i(k)} : t \in [0, T] \} \end{aligned}$$

### 3) 제어기 설계

시스템 (1) 의  $i$  번째 오퍼레이션의 해는 다음과 같이 표현된다.

$$X_{i(k)} = \Phi^k X_{i(0)} + \sum_{j=0}^{k-1} \Phi^{k-j-1} \Gamma U_{i(j)} \quad (4)$$

$X_{i(0)}$ : initial value of  $X_{(k)}$

이 때의 기준 입력에대한 추종 오차는 다음과 같다.

$$\begin{aligned} e_{i(k)} &= X_{d(k)} - X_{i(k)} \\ &= X_{d(k)} - \Phi^k X_{i(0)} - \sum_{j=0}^{k-1} \Phi^{k-j-1} \Gamma U_{i(j)} \end{aligned} \quad (5)$$

또한,  $i+1$  번째 오퍼레이션의 추종 오차도 다음과 같이 표현된다.

$$\begin{aligned} e_{i+1(k)} &= X_{d(k)} - X_{i+1(k)} \\ &= X_{d(k)} - \Phi^k X_{i+1(0)} - \sum_{j=0}^{k-1} \Phi^{k-j-1} \Gamma U_{i+1(j)} \end{aligned} \quad (6)$$

$X_{i(0)} = X_{i+1(0)}$  라고 가정하고,  $i$  와  $i+1$  번째의 관계를 구하기 위하여 식(6)에서 (5)를 뺀다.

$$e_{i+1(k)} - e_{i(k)} = - \sum_{j=0}^{k-1} \Phi^{k-j-1} \Gamma (U_{i+1(j)} - U_{i(j)}) \quad (7)$$

만일,  $i+1$  번째 제어입력을 다음의 적분 형태로 표현한다면

$$U_{i+1}(k) = U_i(k) + \bar{U}_i(t, \Xi_i) \quad (8)$$

여기서,  $\bar{U}_i(t, \Xi_i)$ : additional input

식 (7)은 다음과 같이 표현된다.

$$e_{i+1}(k) = e_i(k) - \sum_{j=0}^{k-1} \Phi^{k-j-1} \Gamma \bar{U}_i(j) \quad (9)$$

#### 4) 제어 법칙

<정 리> : 만일 시스템 (1) 이 안정하고 초기 오차가  $\|e_{i(0)}\|_n \leq \epsilon^*$  을 만족할 때, 다음과 같은 제어법칙은

$$\bar{U}_i(t, \Xi_i) = \delta \Gamma^* (e_{i(k+1)} - \Phi e_{i(k)}) \quad (10)$$

$\delta$  : a positive constant

$\Gamma^*$  : generalized inverse of  $\Gamma$ . ( $= (\Gamma^T \Gamma)^{-1} \Gamma^T$ )

식 (2)의 기준을 유한한 반복작업만에 만족하게한다.

<증 명> :  $i$  와  $i+1$  오퍼레이션간의 관계식 (9)에 제어법칙 (10)을 대입하면 다음 사실을 유도할 수 있다.

$$\begin{aligned} e_{i+1}(k) &= e_i(k) - \sum_{j=0}^{k-1} \Phi^{k-j-1} \Gamma \delta \Gamma^* (e_{i(j+1)} - \Phi e_{i(j)}) \\ &= e_i(k) - \sum_{j=0}^{k-1} (\Phi^{k-j-1} e_{i(k+j)} - \Phi^{k-j} e_{i(k)}) \\ &= e_i(k) - \delta (\Phi^{k-1} e_{i(1)} - \Phi^k e_{i(0)} \\ &\quad + \Phi^{k-2} e_{i(2)} - \Phi^{k-1} e_{i(1)} \cdots + e_{i(k)} - \Phi e_{i(k-1)}) \\ &= (I - \delta) e_i(k) + \delta \Phi^k e_{i(0)} \end{aligned} \quad (11)$$

식 (11)로부터  $0 < \delta < 2$  일때

$$\lim_{i \rightarrow \infty} e_i(k) = \delta \Phi^k e_{i(0)} \quad (12)$$

를 만족한다는 것을 알 수 있다.

또한, 가정에서  $\Phi$ 가 안정하다고 가정하였으므로,

$\delta$  가  $0 < \delta \leq 1$  이고  $\|e_{i(0)}\|_n \leq \epsilon^*$  일때

$$\begin{aligned} \lim_{i \rightarrow \infty} \|X_{d(k)} - X_i(k)\|_n &= \delta \|\Phi^k\| \|e_{i(0)}\|_n \\ &\leq \|e_{i(0)}\|_n \leq \epsilon^* \quad , t \in [0, T] \end{aligned} \quad (13)$$

를 만족한다는 것을 알 수 있다.

또한,  $e_{i(0)} = 0$  이라면 추종 오차는 완전히 0 에 수렴하는 것을 쉽게 알 수 있다.

$$\lim_{i \rightarrow \infty} e_i(k) = 0 \quad , t \in [0, T] \quad (14)$$

위의 결과로 부터  $\delta$  는 수렴성과 안정성에 커다란 영향을 미친다는 것을 알 수 있다. 식 (11)을 보면  $\delta$  가 1 인 경우 2 번째 오퍼레이션 만에 (11)의 마지막 항만이 과도 기간중의 잔류오차로 남는 것을 알 수 있다. 그런데, 이 잔류오차의 크기가  $\delta$  값에 비례하므로 그값을 줄일수록 과도 기간 중의 잔류오차가 감소함을 알 수 있다.

#### 6) 매개변수 추정

반복 학습제어는 미지의 시스템에 적용되므로, 학습의 일환으로서 미지의 매개변수에 대한 추정이 필수적이다. 매개변수 추정 방법은 여러가지 방법이 있으나, 본 연구에서는 RLS (Recursive Least Square) 방법을 도입하였다[4].

제어법칙 식(10)을 발생시키기 위해서는,  $\Phi$  행렬의  $n^2$  개와  $\Gamma$  벡터의  $n$  개를 추정해야 하므로 다음과 같이 행으로 분할하여 한다.

$$\Phi = \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}, \quad \Gamma = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$$

$a_j$  :  $1 \times n$  j th row vector of  $\Phi$

$b_j$  : j th element of  $\Gamma$

또한, 매개변수 벡터  $\theta_j(k)$ 와 데이터벡터  $\phi(k)$ 를 정의하면 식 (1)의 상태 방정식은 다음과 같은  $n$ 개의 식으로 표현할 수 있다.

$$X_{j(k+1)} = \theta_j^T \phi(k), \quad j = 1, \dots, n \quad (15)$$

여기서, 
$$\theta_j^T = [a_j \mid b_j]^T$$
  

$$\phi^T = [X_{j(k)}^T \mid U(k)]$$

$\theta_j(k)$ 의 추정치를  $\hat{\theta}_j(k)$ 로 정의하고 RLS 알고리즘을 정리하면 다음과 같다.

$$\hat{\theta}_j(k+1) = \hat{\theta}_j(k) + G_j(k+1)[X_{j(k+1)} - \hat{\theta}_j^T(k)\phi(k)] \quad (16)$$

$$G_j(k+1) = \frac{P_j(k)\phi(k)}{1/\alpha_t + \phi^T(k)P_j(k)\phi(k)} \quad (17)$$

$$P_j(k+1) = P_j(k) - \frac{P_j(k)\phi(k)\phi^T(k)P_j(k)}{1/\alpha_t + \phi^T(k)P_j(k)\phi(k)} \quad (18)$$

여기서,  $P_j(k)$  : weighting matrix  
 $G_j(k)$  : estimation gain matrix  
 $\alpha(k)$  : forgetting factor

제안된 학습제어 알고리즘에 대한 블록 선도는 Fig. 1과 같다.

### 3. 전산기 시뮬레이션

본 연구에서 제안한 학습제어 알고리즘의 효율성을 고찰하기 위하여 다음의 대상 시스템에 대하여 학습제어 알고리즘을 적용하였다.

<제어대상> : 2차 연속 시스템

$$\dot{X}(t) = \begin{bmatrix} 0 & 1 \\ -600 & -80 \end{bmatrix} X(t) + \begin{bmatrix} 0 \\ 300 \end{bmatrix} U(t)$$

$$Y(t) = [1 \ 0] X(t), \quad X(0) = \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}$$

예제 1) : 다음과 같은 포물선의 추종 경로가 주워졌을 경우에 학습제어 알고리즘을 적용한다.

$$X_{d1} = -0.02 t^2 + 0.25 t + 0.1$$

$$X_{d2} = -0.04 t + 0.25$$

Fig. 2 (a-c)는 이러한 경로를 추종할 경우에 제어법칙 (7)의  $\delta$ 의 변화를 고찰한 결과이다. Fig. 2 (b)를 보면,  $\delta$ 가 1인 경우는 2번째 이터레이션만에 정확하게 원하는 경로를 추종하였다. Fig. 2 (b)는  $\delta$ 가 0.5인 경우인데, 추종 오차가 항상 같은 부호로서 0에 수렴하는 것을 알 수 있다. Fig. 2 (c)는  $\delta$ 가 1.5인 경우인데, 추종 오차의 부호가 교번하는 것을 알 수 있다. 이러한 결과들을 검토해볼 때, 시스템 모델링이 정확하거나 매개변수 추정이 완전하다고 가정하면,  $\delta$ 가 1인 경우 2번째 이터레이션 만에 완전한 제어성능을 기대할 수 있다. 그러나, 매개변수 추정이 어느정도 부정확 할지라도,  $\delta$ 가 0에서 2까지 변할 수 있는 이득 여유 (Gain margine)이 있으므로  $\delta$ 를 1로 정한다면, 제어 안정성을 기대할 수 있을 것이다.

예제 2) 다음의 삼각 파형의 경로에 대하여 (19)의 시스템에 학습제어를 적용한다.

$$X_{d1} = \begin{cases} 0.1 & , 0 \leq t \leq 2.5 \\ 0.4 t - 0.9 & , 2.5 \leq t \leq 5. \\ 9.1 - 0.4 t & , 5. \leq t \leq 7.5 \\ 0.1 & , 7.5 \leq t \leq 10. \end{cases}$$

Fig. 3 (a)(b)는  $\delta$ 를 1로 놓고 학습제어를 적용한 결과이다. 역시 예제 1과 같이 삼각파형의 경로에서도 2번째 이터레이션만에 완전히 추종하는 것을 알 수 있다. Fig. 3 (b)는 이 때의 제어 입력인데 삼각파형의 각 절점에서의 완전한 추종을 위하여 제어 입력이 급격하게 변하는 것을 관찰할 수 있다.

#### 4. 결론

본 연구에서는 미지의 선형 이산 시스템에 대한 반복 학습 제어 알고리즘을 제안하였고, 그 유용성을 시뮬레이션을 통하여 고찰하였다. 본 알고리즘은  $\delta$  가 1 이고 매개변수 추정이 정확한 경우 2번째 이터레이션만에 완전한 추종성능을 보여주었다. 특히, 매개변수 추정이 약간 부정확하다더라도 제안된 알고리즘의 이득 여유 (Gain margine) 가 크기때문에 안정성을 유지할 수 있음을 알수 있었다. 또한, 향후에 시변 시스템과 다변수 시스템으로의 확장이 연구과제로 대두되며 여러가지 공정에 대한 적용이 기대된다.

#### References

- [1] Arimoto, S., Kawamura, S. and Miyazaki, F. : ' Bettering Operation of Robots Learning ', Journal of Robotic Systems, Vol.1, No.2, pp 706-712, 1984
- [2] Furuta, K. and Yamakita, M.: ' The Design of a learning Control System for Multivariable Systems', Proc. IEEE International Symposium on Inteligent Control, PP 371-376, 1987
- [3] Oh, S.R., Bien, Z. and Suh, I.H. : ' An Iterative Learning Control Method with Application for the Robot Manipulators', IEEE Journal of Robotics and Automation. (to be Published)
- [4] Ljung, L. and Soderstrom, T. : ' Theory and Practice of Recursive Identification', MIT Press., London, 1983

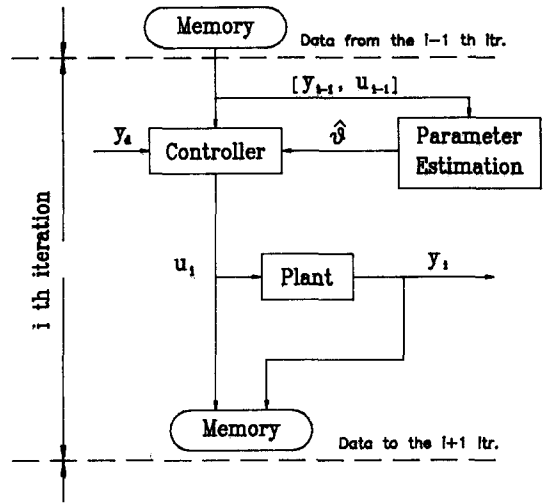


Fig.1 Block diagram of the iterative learning control

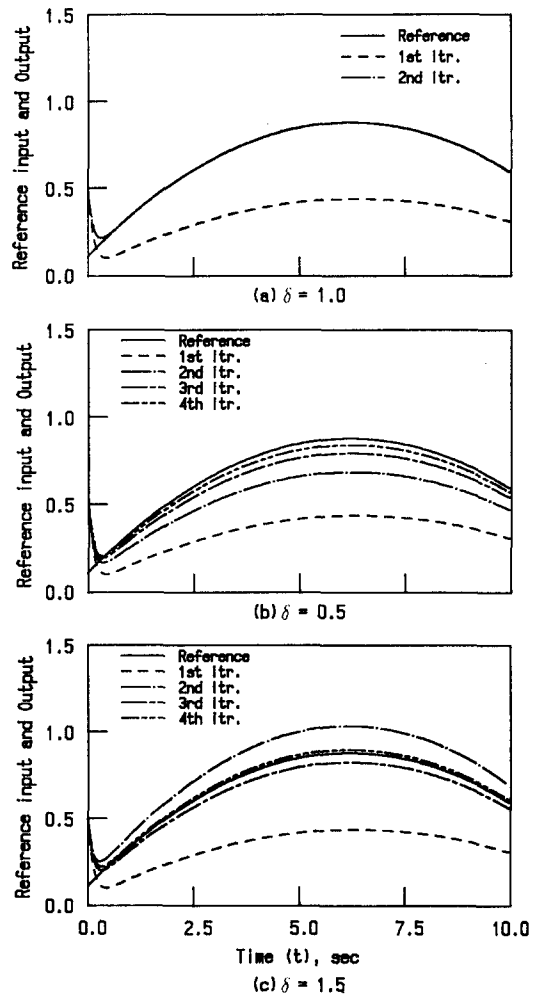
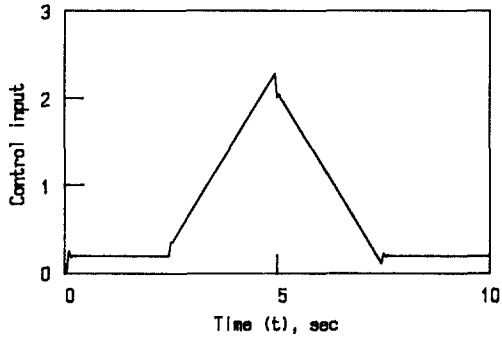
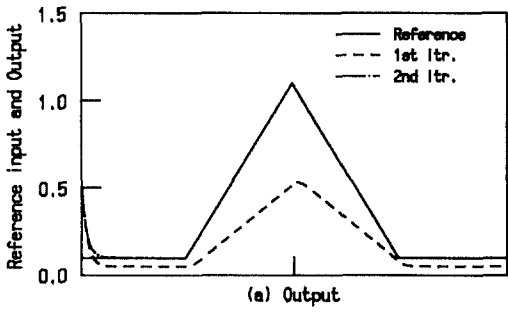


Fig.2 Effect of  $\delta$  in learning control  
Ex.1) Parabolic Input



(b) Control input at 2nd itr.  
 Fig.3 Control input of learning control  
 Ex. 2) Triangular trajectory