An Iterative Learning Control Method with Application for the Robot Manipulator

SANG-ROK OH, ZEUNGNAM BIEN, MEMBER, IEEE, AND IL HONG SUH

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 presented to show the validity of the algorithm. In particular, it is shown that the method is useful for the continuous path control of robot manipulators.

I. Introduction

THERE HAS BEEN a strong belief that "learning," which is one of the most essential characteristics of living animals, can be adopted in engineering systems in order to accomplish intelligent control. In particular, with the recent advent of robotics, a great deal of attention is being paid to the problem of designing a robot controller with a learning capability.

Learning, which is defined in [1] as "changes in the system that are adaptive in the sense that they enable the system to do the same task more efficiently and more effectively the next time," can be employed in engineering systems to achieve a better control via knowledge acquisition and/or skill refinement [2]. Especially, the concept of learning for skill refinement through practice is believed to have a variety of applications for autonomous systems.

The conventional adaptive control system may be considered as a type of learning control system in which achieving regulation, for example, is the objective of skill refinement. In this case, however, the objective is acquired in one operation and the type of learning should be simple. For instance, the adaptive control system employed for a robot manipulator with unknown dynamics may achieve asymptotic positioning control but may not be capable of tracking a given entire span of a trajectory within a specified error bound since a conventional adaptive control law results in asymptotic convergence and, as such, initial and/or intermediary performance may not be strictly controlled [3], [4].

Another type of learning control systems discussed in the literature [5]–[8] proposes learning for skill refinement which is accomplished through repetition or a sequence of iterative operations. If the skill or the objective performance to learn is not simple, the iterative method would naturally be a better alternative.

Manuscript received November 18, 1986; revised February 24, 1988.

S.-R. Oh is with the Power Controls Laboratory, Korea Advanced Institute of Science and Technology, Seoul, Korea.

Z. Bien is with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Seoul, Korea.

I.-H. Suh is with the Department of Electronics Engineering, Hanyang University, Seoul, Korea.

IEEE Log Number 8821749.

In this paper, we will present a method of learning control which can be used for the control of a robot manipulator with path-dependent tasks. To be specific, let us consider the linear periodic system described by

$$\dot{x}(t)=A(t)x(t)+B(t)u(t) \tag{1}$$

$$x(0)=\xi^0 \tag{2}$$

where x and u are an $n \times 1$ state vector and $m \times 1$ control vector, respectively. The $n \times n$ matrix function $A(t)$ and the $n \times m$ matrix function $B(t)$ are assumed to contain uncertain parameters but are known to be continuous and periodic with period T such that

$$A(t+T)=A(t)$$
$$B(t+T)=B(t). \tag{3}$$

Now consider the following problem (P).

Problem (P): Let $x^D(t)$, $0 \le t \le T$ denote the given desired state trajectory. Let $\epsilon^* > 0$ be a given tolerance bound. Find a control function $u(t)$, $0 \le t \le T$ such that the corresponding state trajectory $x(t)$ of the linear system in (1) with initial condition in (2) satisfies

$$p(x(t))=\|x(t)-x^D(t)\|_n \le \epsilon^*, \qquad 0 \le t \le T \tag{4}$$

where $\|\cdot\|_n$ denotes the n-dimensional maximum norm.

In the problem, the inequality constraint of (4) is difficult to handle and as a consequence, no conventional adaptive control method can guarantee to yield a solution for the problem. There are available, however, some iterative learning control methods. Specifically, Raibert [5] proposed a learning control method using a parameterization technique. But the parameter learning technique in Raibert's method shows limited application due to difficulty in frequently computing the inverse of a matrix with rank deficiency. Also Arimoto *et al.* [6]–[8] proposed simple iterative learning control strategies with a proof of convergency. Their formulations require a rather specific knowledge of the manipulator dynamics in determining the controller gain matrix, and thus the method can be unsuccessful if the controller gain matrix is not properly chosen.

In this paper, we propose a learning control method which may be considered as a modified version of the method of Arimoto in [6]. More specifically, we first adopt a parameter estimator to alleviate the difficulty of determining *a priori* the controller gain $\Gamma(t)$ in [6]. Special attention should be given to the fact that parameter estimation is performed in the domain of iterative sequence of operation with time frozen, not with respect to the continuous time domain as is done in conventional adaptive control techniques [3], [4]. Then the next control sequence is synthesized by adding a compensation signal generated from the estimated inverse model to the current control. A sufficient condition for the convergency is provided and two simple numerical examples are given. In particular, the proposed method is shown to be applicable for the motion control of a robot manipulator, which is illustrated for a two-link robot manipulator.

In the sequel, given a matrix B, B^T denotes the transpose of B and B^+ implies the generalized inverse of B [12], respectively. 0_n and I_n denote the $n \times n$ null matrix and the $n \times n$ identity matrix, respectively, and diag $\{\cdot\}$ denotes the diagonal matrix. $C^n[0, T]$ is used to represent the normed linear space of all continuous n-vector functions on $[0, T]$ with sup-norm topology. Also the normed linear space $D^n[0, T]$ (see [9]) consists of all n-vector functions on $[0, T]$, which are continuous and have continuous derivatives with the norm defined as

$$\|x\| = \max_{0 \le t \le T} \|x(t)\|_n + \max_{0 \le t \le T} \|\dot{x}(t)\|_n .$$

II. An Iterative Learning Control Method for a Class of Linear Periodic Systems

Regarding the problem (P) stated in the previous section, we will seek a solution by an iterative learning method. For this, let the suffix k denote the iteration number of operation such that, for example, $x_k(t)$ is the value of the system state at time t, $0 \le t \le T$, at the kth operation, etc. The notations $e_k(t)$ and $u_k(t)$ will be similarly defined.

We will first propose a parameter estimation scheme employing a recursive least squares method in which the recursion formula works in the domain of iteration sequence of operations with time frozen, not with respect to the continuous time domain t as can usually be seen in the literature (e.g. [10]). Since the system matrices are assumed to be periodic matrices (see (3)), we have that for any fixed time $\hat{t} \in [0, T]$, $A(\hat{t}) = A(\hat{t} + kT)$ and $B(\hat{t}) = B(\hat{t} + kT)$ for $k = 0, 1, 2, \cdots$. So, for each $\hat{t} \in [0, T]$, let $\tilde{A}_k(\hat{t})$ and $\tilde{B}_k(\hat{t})$ denote the modeled matrices of $A(\hat{t})$ and $B(\hat{t})$ at the kth operation, respectively. Then for each $\hat{t} \in [0, T]$, $\tilde{A}_k(\hat{t})$ and $\tilde{B}_k(\hat{t})$ can be obtained by the recursive least squares method [10] using the data $\tilde{A}_l(\hat{t})$ and $\tilde{B}_l(\hat{t})$, for $l = 0, 1, \cdots, k-1$, and the input and output values at $\hat{t} + hT$, $h = 1, 2, \cdots, k$. To present a formula more specifically, let the matrices $A(t)$ and $B(t)$ in (1) be rewritten as

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

where for each $i = 1, 2, \cdots, n$, $a^i(t)$ and $b^i(t)$ are an $1 \times n$ and $1 \times m$ row vector, respectively. Define the $1 \times (n+m)$ vectors $\theta^i(t)$ and $\psi(t)$ as follows:

$$\theta^{i^T}(t) \triangleq [a^i(t) \; b^i(t)]$$
$$\psi(t) \triangleq [x^T(t) \; u^T(t)]^T. \tag{5}$$

For each $\hat{t} \in [0, T]$, we wish to identify n^2 parameters of $A(\hat{t})$ and nm parameters of $B(\hat{t})$. For this, let

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

such that the ith component of $y(\hat{t})$ is given by

$$y^i(\hat{t}) = \dot{x}^i(\hat{t}), \qquad i = 1, 2, \cdots, n.$$

Then the system in (1) can be described by

$$y^i(\hat{t}) = \theta^{i^T}(\hat{t})\psi(\hat{t}), \qquad \text{for } i = 1, 2, \cdots, n. \tag{7}$$

Now, we may apply a recursive least squares parameter identification algorithm with regards to the relation of (7). That is, for each $\hat{t} \in [0, T]$ and for $i = 1, 2, \cdots, n$, let the estimated parameter vector $\tilde{\theta}^i_k(t)$ in the kth operation be given by

$$\tilde{\theta}^i_k(\hat{t}) = \tilde{\theta}^i_{k-1}(\hat{t}) + F^i_k(\hat{t})[y^i_k(\hat{t}) - \tilde{\theta}^{i^T}_{k-1}(\hat{t})\psi_k(\hat{t})] \tag{8}$$

where

$$F^i_k(\hat{t}) = \frac{S^i_{k-1}(\hat{t})\psi_k(\hat{t})}{1/\alpha^i_k(\hat{t}) + \psi^T_k(\hat{t})S^i_{k-1}(t)\psi_k(\hat{t})} \tag{9}$$

and

$$S^i_k(\hat{t}) = S^i_{k-1}(t) - \frac{S^i_{k-1}(\hat{t})\psi_k(\hat{t})\psi^T_k(\hat{t})S^i_{k-1}(\hat{t})}{1/\alpha^i_k(\hat{t}) + \psi^T_k(\hat{t})S^i_{k-1}(\hat{t})\psi_k(\hat{t})}. \tag{10}$$

Here for each $\hat{t} \in [0, T]$, initial parameters $\tilde{\theta}^i_0(\hat{t})$, $i = 1, \cdots, n$ and initial weighting matrices $S^i_0(\hat{t})$, $i = 1, \cdots, n$ are given appropriately as in the conventional recursive least squares estimation algorithm [10]. Also $\alpha^i_k(\hat{t})$, $i = 1, \cdots, n$, the forgetting factors, are suitably chosen in such a way that $0 < \alpha^i_k(\hat{t}) \le 1$. It is remarked that if the measurement data $y(t)$ in (8) are noise-corrupted, the above estimation method may be inappropriate for use in general and some modified scheme may be necessary. In case of robot manipulators, $y(t)$ can be measured by tachometers and accelerometers. It is also remarked that the above estimation scheme is essentially the same in discrete time as given in [10, pp. 16–22].

To present an iterative learning control method, let $\psi(t)$ be the unique solution of the fundamental matrix differential equation described by

$$\dot{\Psi}(t) = A(t)\Psi(t) \quad \Psi(0) = I_n. \tag{11}$$

Now the algorithm of the proposed learning controller is given below.

Algorithm: Let $\epsilon^* > 0$ be given as the state trajectory error bound. Let the initial control $u_0(t)$, $0 \le t \le T$, be given as an m-vector continuous function. Also let the initial modeled system matrices $\tilde{A}_0(t)$ and $\tilde{B}_0(t)$, $0 \le t \le T$, be given as continuous matrices on $[0, T]$. Set $k = 0$.

Step 1: Let

$$e_k(t) = x^D(t) - x_k(t), \qquad 0 \le t \le T. \tag{12}$$

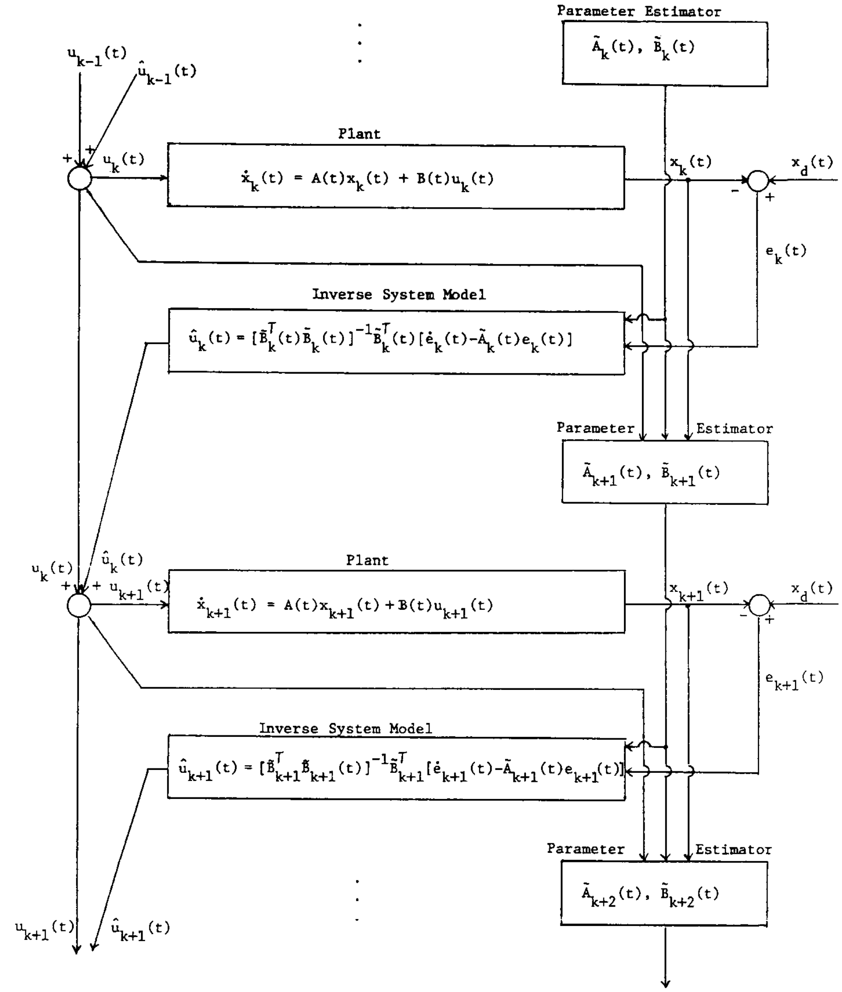


Fig. 1. A schematic diagram of the proposed learning control.

Here $x_k(t)$ is the solution of (1) and (2), i.e.,

$$x_k(t)=\Psi(t)x_k(0) + \int_0^t \Psi(t)\Psi^{-1}(\tau)B(\tau)u_k(\tau)\,d\tau. \tag{13}$$

Step 2: Let

$$\hat{u}_k(t)=\tilde{B}_k^+(t)[\dot{e}_k(t)-\tilde{A}_k(t)e_k(t)]. \tag{14}$$

Step 3: Let

$$u_{k+1}(t)=u_k(t)+\hat{u}_k(t). \tag{15}$$

Step 4: If $\|e_k(t)\|_n \leq \epsilon^*$, stop.
Otherwise, set $k = k + 1$, and go to Step 1.

The structure of the above algorithm is schematically shown in Fig. 1.

Several remarks are in order. First, it is noted that (14) may be interpreted as a kind of an inverse system model of (1) in the sense that in (1) $u_k(t)$ generates $x_k(t)$ through the matrix functions $A(t)$ and $B(t)$ while in (14) $e_k(t) = x^D(t) - x_k(t)$ together with $\dot{e}_k(t)$ generates $\hat{u}_k(t)$ through $\tilde{A}_k(t)$ and $\tilde{B}_k(t)$. Secondly, if the rank of $B(t)$ is greater than or equal to unity, a generalized inverse of $B(t)$ always exists, and moreover if the rank of $B(t)$ is m, the generalized inverse of $B(t)$ is given by

$$B^+(t)=[B^T(t)B(t)]^{-1}B^T(t)$$

which is the right inverse of $B(t)$ [12]. Finally, it is remarked that even though parameter estimation should be conducted for each $\hat{t} \in [0, T]$, a practical estimator implements the algorithm only for a finite but sufficiently large number N of times, and $\hat{t}$ may be chosen to be $\hat{t} = jT/N, j = 0, \cdots, N$ as in [6].

The above algorithm is shown to be convergent in the following statement.

Theorem 1

Consider the linear periodically time-varying system in (1). If the system in (1) is totally stable and if the parameter estimation scheme given by (8)–(10) is convergent, then the iterative learning controller in (12)–(15) with $x_k(0) = x^D(0) = \xi^0$ for $k = 0, 1, \cdots$, yields

$$\lim_{k\to\infty} \|e_k\| = 0.$$

Proof of Theorem 1: Define the linear operator $L:C^m[0, T] \rightarrow D^n[0, T]$ by

$$[L(u)](t) = \int_0^t \Psi(t)\Psi^{-1}(\tau)B(\tau)u(\tau)\, d\tau, \quad \text{for each } u \in C^m[0, T]. \tag{16}$$

Also let $P:D^n[0, T] \rightarrow C^m[0, T]$ be defined by

$$[P(e_k)](t) = B^+(t)[\dot{e}_k(t) - A(t)e_k(t)]. \tag{17}$$

Further, introduce the operator $\tilde{P}_k:D^n[0, T] \rightarrow C^m[0, T]$ for (14) such that

$$[\tilde{P}_k(e_k)](t) = \tilde{B}_k^+(t)[\dot{e}_k(t) - \tilde{A}_k(t)e_k(t)]. \tag{18}$$

Then the variation of constant's formula of (13) and (16) implies

$$x_k(t) - \Psi(t)\xi^0 = (Lu_k)(t).$$

Also (14) and (18) yield

$$\hat{u}_k(t) = (\tilde{P}_k e_k)(t)$$

while (16) and (17) result in

$$[L(P(e_k))](t) = e_k(t).$$

Now, observe that

$$\begin{aligned}\|e_{k+1}\| &= \|x_{k+1} - x^D\| \\ &= \|Lu_{k+1} + \Psi\xi^0 - x^D\| \\ &= \|x_k - x^D + L\hat{u}_k\| \\ &= \|L\tilde{P}_k e_k - LPe_k\| \\ &\le \|L\|\ \|\tilde{P}_k e_k - Pe_k\| \\ &\le \|L\|\ \|\tilde{P}_k - P_k\|\ \|e_k\|.\end{aligned}$$

Since the system in (1) is assumed to be totally stable, we find that L is a bounded operator [13], i.e., there is an $M > 0$ such that $\|Lu\| \le M\|u\|$ for every $u \in C^m[0, T]$. Further, since the parameter estimation scheme is convergent by assumption, there exists an integer N for a given number α, $0 < \alpha < 1$, such that

$$\|\tilde{P}_k - P\| \le \alpha/M, \qquad \text{for all } k \ge N.$$

Therefore, we have

$$\|e_{k+1}\| \le \alpha\|e_k\|, \qquad \text{for } k \ge N \text{ and } 0 < \alpha < 1.$$

This completes the proof.

It is recalled here that the above convergence proof is based on the assumption that $x_k(0) = x^D(0)$ for all k as in [6]. Since updating the initial settings may take some time in reality, one may argue that such an assumption is hardly fulfilled for linear periodic systems continuously operating in time. But we may say updating initial conditions can be done in robotic applications or in other repetitive manufacturing operations, provided that a "repositioning scheme" is used at the end of each period.

III. Numerical Examples

To show some use of the proposed learning control algorithm, two numerical examples are illustrated: one for a single-input single-output linear time-varying system and the other for a two-link robot manipulator. Also in Example 1, the performance of the proposed algorithm is compared with the algorithm in [6] via a digital computer simulation.

Example 1: A Linear Time-Varying System

Consider the linear time-varying system whose dynamics is described by

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

$$\begin{bmatrix}x^1(0)\\ x^2(0)\end{bmatrix} = \begin{bmatrix}0\\ 0\end{bmatrix}, \qquad 0 \le t \le 1. \tag{19}$$

Let the desired state trajectories $x^{D^i}(t)$, $i = 1, 2$ for $t \in [0, 1]$ be given by

$$\begin{aligned}x^{D^1}(t) &= t^3(4-3t)\\ x^{D^2}(t) &= 12t^2(1-t)\end{aligned} \tag{20}$$

and the tolerance bound $\epsilon^* = 0.005$. To be exact, the variable t of matrix $A(t)$ in (19) and the right-hand side of (20) should be replaced by $t - [t]$, where $[t]$ is the Gaussian number of t, such that the system could be periodic with $T = 1$. Also, it is not difficult to show that the system in this example is totally stable.

Now, to apply the learning control method proposed in the previous section, let the initial input $u_0(t)$ and the initial system matrices $\tilde{A}_0(t)$ and $\tilde{B}_0(t)$, for $t \in [0, 1]$ be given by

$$\begin{aligned}u_0(t) &= 0, \qquad 0 \le t \le 1\\ \tilde{A}_0(t) &= \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\\ \tilde{B}_0(t) &= \begin{bmatrix}0\\ 0.5\end{bmatrix}, \qquad 0 \le t \le 1.\end{aligned} \tag{21}$$

Also, let the initial conditions for the parameter estimation in (8)–(10) be given as

$$\begin{aligned}&\alpha_k^i(t) = 1, \qquad i = 1, 2, \quad k = 0, 1, 2, \cdots, \quad 0 \le t \le 1\\ &S_0^i(t) = \text{diag}\,\{10000,\ 10000,\ 10000\}, \qquad 0 \le t \le 1.\end{aligned} \tag{22}$$

Then we can obtain $x_k^i(t)$, $i = 1, 2$, for the kth iteration, as shown in Fig. 2. It is noted from Fig. 2 that even in the case of relatively large initial parameter discrepancies as given in (19) and (21), we can observe that five iterative operations are sufficient to generate a trajectory which is within ϵ^* bound of the desired state trajectory $x^{D^i}(t)$, $i = 1, 2$, respectively.

To compare the effectiveness of our learning control

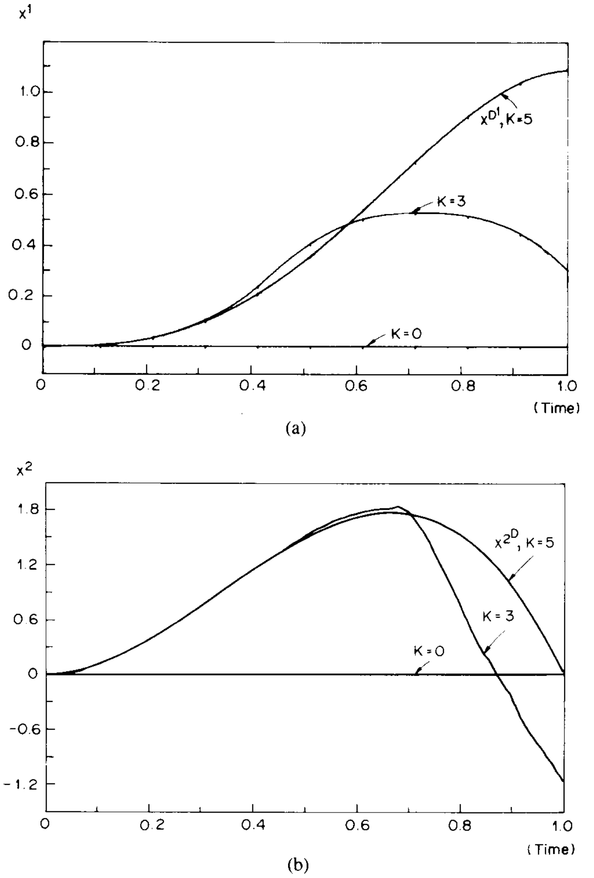


Fig. 2. State trajectories by the proposed learning algorithm. (a) $X^1(t)$ trajectories. (b) $X^2(t)$ trajectories.

algorithm with the one proposed by Arimoto *et al.* [6], recall that Arimoto's algorithm is given by

$$y_k(t)=C(t)\Psi(t)x_k(0) + \int_0^t C(t)\Psi(t)\Psi^{-1}(\tau)B(\tau)u_k(\tau)\,d\tau \tag{23}$$

$$e_k(t)=y^D(t)-y_k(t) \tag{24}$$

$$u_{k+1}(t)=u_k(t)+\Gamma(t)\dot{e}_k(t) \tag{25}$$

where y is an r-dimensional output vector and C is an $r \times n$ output matrix. With respect to (23)-(25), the convergence condition given by Arimoto is

$$\|I_r - C(t)B(t)\Gamma(t)\| < 1. \tag{26}$$

Obviously, if the system dynamics contain uncertain parameters in $B(t)$, some difficulties may arise in choosing $\Gamma(t)$ in (25) in such a way that the inequality (26) is satisfied. Nevertheless, for comparison, we first applied Arimoto's algorithm to the system (19) under the assumption that the dynamics is completely known. The gain $\Gamma(t)$ was chosen to be unity to get the best convergence rate and, via simulation, we found the maximum error to be $0.06 > \epsilon^*$ after ten iterations, as shown in Fig. 3.

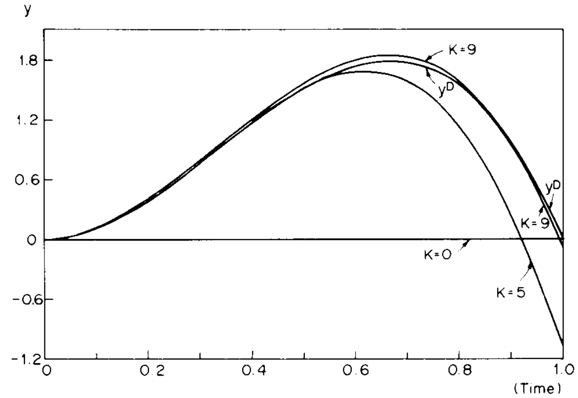


Fig. 3. Output trajectories by Arimoto's algorithm.

Next, we applied Arimoto's algorithm for the system with $B(t)$ replaced by $\bar{B}_0(t)$ in (21) to simulate the case when the dynamics are unknown. In this case, $\Gamma(t)$ was chosen to be 2 to render the best convergence rate for the assumed model, but again the maximum absolute error between the desired and the output trajectories was 0.21 after ten iterations. At any rate, in this particular example, we observed that the learning algorithm proposed in this paper showed a better convergence rate than that of Arimoto's [6], without the difficulty of determining $\Gamma(t)$ when the system dynamics are not completely known.

Example 2: Application for Continuous-Path Control of Robot Manipulators

It is a common practice that, for continuous-path control of an n-joint robot manipulator, the desired nominal trajectory is given at the planning stage and the nominal input torque is computed from the $N - E$ equations of motion with regards to the given nominal trajectory. If this computed input torque is used, the motion trajectory of the manipulator usually falls in a neighborhood of the given desired trajectory. It is in general difficult to obtain exact tracking due to the difficulties in modeling the robot dynamics including frictions, backlash, and nonrigidity. To eliminate the resulting path error, the additional torque input is required and may be supplied by a control based on a linear system model. This is where the proposed iterative learning controller can be used. To see this, first let us note that the equation of motion of the manipulator linearized in the neighborhood of the given desired trajectory can be represented as a linear time-varying system [3]. Secondly, when the robot manipulator moves along the same trajectory in the work space for repetitive operation, the system can be considered as a periodic system. In addition, if the manipulator is legitimately assumed to contain typical nonlinearities such as inertial, centrifugal and Coriolis, and gravity types only, the coefficient matrices are continuous functions of time. We may now invoke the Floquet theory on the equivalence of a linear periodic system to a linear constant system [13] to conclude that the linearized system can be made totally stable with appropriate feedback if needed. Thus the iterative learning control algorithm in (12)-(15) can be applied

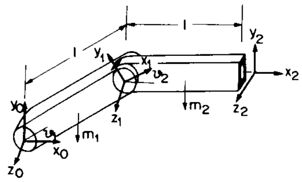


Fig. 4. A two-link robot arm.

TABLE I
INPUT DATA FOR SIMULATION

Parameter \ Value	True Value	Modeled 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

to the linearized system with the parameter estimator in (8)–(10) with appropriate dimensions.

To show use of the algorithm for the continuous path control of robot manipulators, a digital computer simulation is performed with a two-link robot manipulator as shown in Fig. 4, whose dynamic equations of motion are given in the following [11]:

$$\begin{bmatrix}\tau^1\\ \tau^2\end{bmatrix}=\begin{bmatrix}\frac{1}{3}m_1l^2+\frac{4}{3}m_2l^2+m_2c_2l^2 & \frac{1}{3}m_2l^2+\frac{1}{2}m_2c_2l^2\\ \frac{1}{3}m_2l^2+\frac{1}{2}m_2c_2l^2 & \frac{1}{3}m_2l^2\end{bmatrix}\begin{bmatrix}\ddot{\theta}^1\\ \ddot{\theta}^2\end{bmatrix}+\begin{bmatrix}\frac{1}{2}m_2s_2l^2\dot{\theta}^{2^2}-m_2s_2l^2\dot{\theta}^1\dot{\theta}^2\\ \frac{1}{2}m_2s_2l^2\dot{\theta}^{1^2}\end{bmatrix}+\begin{bmatrix}-\frac{1}{2}m_1glc_1-\frac{1}{2}m_2glc_{12}-m_2glc_1\\ -\frac{1}{2}m_2glc_{12}\end{bmatrix}. \quad (27)$$

Here c_i, s_i, and c_{ij} are defined by $\cos\theta^i$, $\sin\theta^i$, and $\cos(\theta^i+\theta^j)$, respectively, θ^1 and θ^2 are joint variables, and the link masses and lengths for the simulation are given in Table I. Note that (27) is the standard form of the L-E equation of motion given by

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

Let the desired trajectories for $t\in[0,1]$ and tolerance bound ϵ^* be given by

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

$$\theta^{D^2}(t)=\frac{5}{3}t^3-\frac{5}{2}t^2-\frac{5}{6} \quad (28)$$

and

$$\epsilon^*=1.0\ [\text{deg}]=0.017\ [\text{rad}]. \quad (29)$$

To apply the algorithm in (12)–(15) for the two-link manipulator, first we obtain the nominal torque input $\tau^N(t)$ by employing the Newton–Euler equation [11] with the modeled parameters given in Table I. And for the iterative learning controller, an initial input is chosen as zero for each $t\in[0,1]$. Also we let $\tilde{A}_0(t)$ and $\tilde{B}_0(t)$ as

$$\tilde{A}_0(t)=\begin{bmatrix}0_2 & I_2\\ 0_2 & 0_2\end{bmatrix}$$

$$\tilde{B}_0(t)=\tilde{D}^{-1}(\theta^D(t)),\qquad t\in[0,1] \quad (30)$$

where $\tilde{D}(\theta)$ denotes the modeled inertia matrix obtained for $D(\theta)$ which, we assume, contains uncertain parameters. Further, let the initial conditions for the parameter estimation scheme in (8)–(10) be given as

$$\alpha_k^i(t)=0.93,\qquad i=1,2,\quad k=0,1,2,\cdots,\quad 0\le t\le 1$$

and

$$S_0^i(t)=\text{diag}\{100000,\cdots,100000\},\qquad 0\le t\le 1. \quad (31)$$

Simulation results for the variables $\theta_k^i(t)$ for $i=1,2$, and for the kth iterative operation are shown in Fig. 5. It is observed from Fig. 5 that even in the case of considerable parameter uncertainties, as in Table I, twenty iterative operations reduce the maximum absolute error to less than ϵ^* bound for $i=1,2$ and for every $t\in[0,1]$.

IV. Concluding Remarks

An iterative learning control algorithm for a class of linear periodic systems was proposed in which parameter estimation was performed in the domain of iterative sequence of operations with the time frozen. A sufficient condition for the convergency of the proposed algorithm was given. Also it was shown that the proposed learning controller could be applied

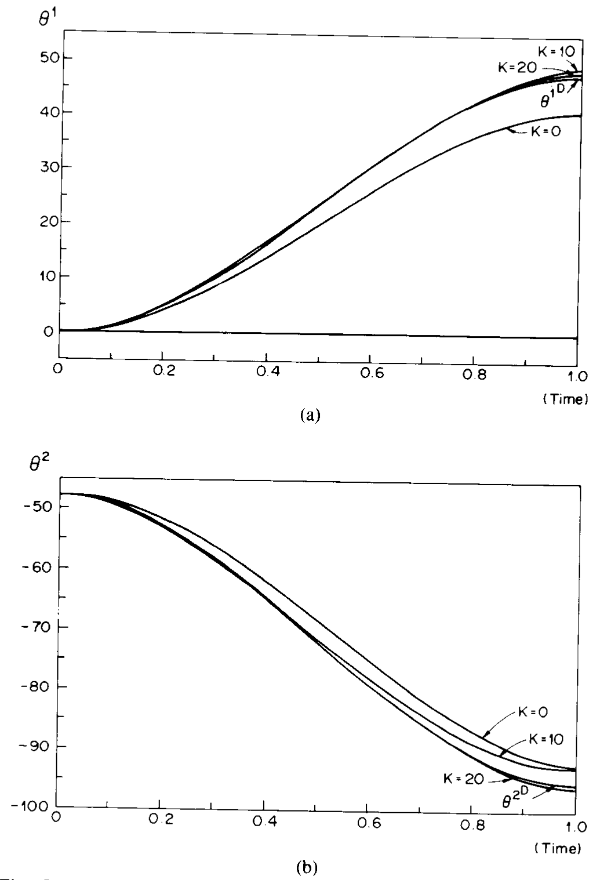


Fig. 5. Joint angle trajectories by the proposed learning algorithm. (a) Joint angle 1. (b) Joint angle 2.

for the continuous path control of robot manipulators. It is remarked that the proposed algorithm works reasonably well only for the case of small perturbations with respect to a nominal trajectory. Further study should be directed towards the elimination of the assumption on updated initial conditions and generality of the algorithm for a robot in an environment with large parameter uncertainties and perturbations.

Acknowledgment

The authors would like to thank the reviewers for their valuable comments in revision of the paper.

References

[1] H. A. Simon, "Why should machine learn?" in *Machine Learning*. Berlin, Heidelberg, New York: Springer-Verlag, 1984, pp. 25-38.

[2] J. G. Carbonnel, R. S. Michalski, and T. M. Mitchell, "An overview of machine learning," in *Machine Learning*. Berlin, Heidelberg, New York: Springer-Verlag, 1984, pp. 3-23.

[3] C. S. G. Lee, M. J. Chung, and B. H. Lee, "An approach of adaptive control for robot manipulators," *J. Robotic Syst.*, vol. 1, no. 1, 1984

[4] A. J. Koivo and T.-H. Guo, "Adaptive linear controller for robotic manipulators," *IEEE Trans. Automat. Contr.*, vol. AC-28, no. 2, pp. 162-171, 1983.

[5] M. H. Raibert, "Motor control and learning by the state space model," Ph.D. dissertation, Dept. of Psychology, MIT, 1977.

[6] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *J. Robotic Syst.*, vol. 1, no. 2, 1984.

[7] ——, "Can mechanical robots learn by themselves?" in *Proc. 2nd Int. Symp. on Robotics Research* (Kyoto, Japan, Aug. 1984).

[8] S. Kawamura, F. Miyazaki, and S. Arimoto, "Iterative learning control for robotic systems," in *Proc. IECON'84* (Tokyo, Japan, 1984).

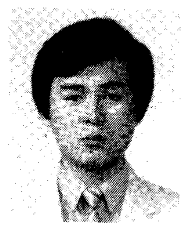
[9] D. G. Luenberger, *Optimization by Vector Space Methods*. New York, NY: Weily, 1969.

[10] L. Ljung and T. Soderstrom, *Theory and Practice of Recursive Identification*. London, England: MIT Press, 1983.

[11] C. S. G. Lee, R. C. Gonzalez, and K. S. Fu, *Tutorial on Robotics*. Washington, DC: IEEE Computer Society Press, 1984.

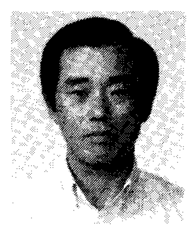
[12] S. Barnett, *Matrices in Control Theory*. New York, NY: Van Nostrand Reinhold, 1971.

[13] C. T. Chen, *Linear System Theory and Design*. New York, NY: Holt, Rinehart and Winston, 1984.



Sang-Rok Oh was born in Seoul, Korea. He received the B.S. degree in electronics engineering from Seoul National University, Seoul, in 1980, and the M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Seoul, in 1982 and 1987, respectively.

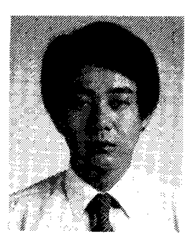
He worked as a Research Associate at the Systems Control Laboratory of KAIST, conducting the design and implementation of multiprocessor-based automatic assembly machine for microelectronic components and robotic control system for multi-legged locomotion. He is currently Senior Research Engineer at the Power Control Laboratory of KAIST. His research interests include intelligent control of robot manipulators, machine vision, applications of modern control theory to industrial processes, and factory automation.



Zeungnam Bien (S'72-M'75) received the B.S. degree in electronics engineering from Seoul National University, Seoul, Korea, in 1969 and the M.S. and Ph.D. degrees in electrical engineering from the University of Iowa, Iowa City, Iowa, in 1972 and 1975, respectively.

He is currently Professor of Electrical Engineering at the Korea Advanced Institute of Science and Technology in Seoul, Korea. His research interests include intelligent control, robotics, and applications of modern control theory to industrial processes. Prior to joining the Institute in 1977, he taught at the Department of Electrical Engineering, University of Iowa. He worked as visiting faculty at the University of Iowa during his 1981-1982 sabbatical year, and as visiting researcher at CASE Center of Syracuse University, Syracuse, NY, for 6 months starting in September, 1987.

Dr. Bien is a member of the Korea Institute of Electrical and Electronics Engineers, Japan Industrial Robot Association and Eta Kappa Nu.



Il Hong Suh was born in Seoul, Korea. He received the B.S. degree in electronics engineering from Seoul National University. Seoul, in 1977, and the M.S. and Ph.D. degrees in electrical engineering (EE) from Korea Advanced Institute of Science and Technology (KAIST), Seoul, in 1979 and 1982, respectively.

From 1982 to 1985, he was a Senior Research Engineer at the Technical Center of Daewoo Heavy Industries, Ltd., Inchon, Korea, where he was involved in the research on machine vision and the development of Daewoo NOVA10 robot controller. During 1985-1986, he has has been a Senior Researcher for an automation-related Korea National Project at the Systems Control Laboratory of KAIST. In 1987, he was a Visiting Research Scientist at the Robotics Division, CRIM, of the University of Michigan, Ann Arbor. Since 1985, he has been an Assistant Professor of the Department of Electronics Engineering, Hanyang University, Seoul, Korea. His research interests include intelligent control of robot manipulators, coordination of multiple robot arms, machine vision, and applications of advanced control theory to industrial systems.