J.Basrah Researches (Sciences) Vol. 33. No.2.(112 -121) Jun. (2007)

A STUDY ON THE HIGHER ORDER LEARNING CONTROL ALGORITHM

Khulood Moosa Omran[1], Turki Yunis Abdulla[2] and Awatiff Salman Algam[3]

[1] Arab Gulf Studies Center , Basrah University
[2] Electrical Engineering Department , Basrah University
[3] Computer system department,Technical Institute of Basra

ISSN –1817 –2695
(Accepted 24/1/1999)

Abstract

Learning control is a new approach to the problem of improving transient response over fixed time duration by iterative training.
It is a recursive on line control method that relies on less calculation and requires less a priori knowledge about the system dynamics. It can be used as a real time control method for complex systems such as robot manipulators.
In this paper the proposed algorithm is used for improving the performance at the next trail on the basis of more than one past error history contained in the trajectories generated at prior iterations. A convergence proof is given and examples are provided to show the effectiveness of the algorithm.

1.Introduction

Recently a novel control method called “learning control” is receiving attention as a method for controlling uncertain dynamic systems in a simple manner. The idea is to apply a simple algorithm repetitively to unknown plant until perfect tracking is achieved. It is applicable for the control of robot manipulators with repetitive tasks. It is simple to calculate and independent of system dynamics. It is a new approach for the control problem of skill refinement. We humans are able to acquire the skill via a long series of repeated exercises. Motivated by this observation, much literature concerned with learning control techniques for robotic systems has accumulated such as Arimoto et al [1,2,3,4,5], Craig [6] ,Bondi et al [7], Z. Bien et al[8], mainly during the past several years.
Learning control is considered to be a mathematical model of motor program learning for skilled motions in the central nervous system[4]. As it is now well known , such as techniques are based on the use of repeated trials of tracking of a pre assigned trajectory at each trail, the position , velocity and acceleration errors with respect to the reference, are recorded to be exploited by a suitable algorithm in the next trail in the aim of progressively reducing the trajectory error. Most of recent papers are concerned with only a class of tasks described in terms of joint trajectory tracking and the control signal is synthesized from only one single history data.
In other words, for a desired motion given to a robot arm in which its end-point is free to move in the three dimensional space, the arm repeats exercises to reduce the trajectory tracking errors depending on only one single history data. Gradually and eventually can learn to exactly trace the given desired motion. To improve the robustness of the learning control method, a higher- order learning control algorithm were proposed [8,9]. We develop here a new algorithm which has the following new features:

1. It is a general algorithm, which can be applied for all types of systems.
2. It gives a clear illustration of how the algorithm can be applied and shows that it needs only some information about the amount of coupling between input and output (position ,velocity or acceleration).

3. We call this algorithm a general third order algorithm because the control inputs in each trail is synthesized from three history data e.g.
$u_{k+1} = F(u_k, u_{k-1}, u_{k-2}, e_k, e_{k-1}, e_{k-2})$.

2. GENERAL THIRD-ORDER LEARNING CONTOL ALGORITHM

Consider the linear time – invariant dynamical system:

$$\dot{x}(t) = A\,x(t) + Bu(t), \qquad x(0) = x_o \qquad \text{.......(1)}$$
$$y(t) = C\,x(t) + D\,u(t)$$

where $x(t)$ and $u(t)$ denote an nx1 state vector and a px1 control vector respectively, and y is an mx1 output vector. A , B , C and D are constant matrices with appropriate dimensions.

Let $y_d(t)$, $0 \le t \le T$, be the desired output trajectory and $\varepsilon > 0$ be given as a tolerance bound , under the condition that the matrices A, B, C and D are not fully known , we wish to find a control function $u(t)$, $0 \le t \le T$, such that the corresponding output trajectory $y(t)$ of the linear system in eq. (1) satisfies
$e(y(t)) = \| y_d(t) - y(t) \| \le \varepsilon$, $0 \le t \le T$
As a solution method , we proposed a new type of third order iterative learning control of the form

$$u_{k+1}(t) = A_1 u_k(t) + A_2 u_{k-1}(t) + A_3 u_{k-2}(t) + B_1 e_k(t) + B_2 e_{k-1}(t) + B_3 e_{k-2}(t) \ldots(2)$$

where $e_k(t) = y_d(t) - y_k(t)$

Let the suffix k denotes the iteration ordinal number of trial, so that , for example $y_k(t)$
Is the value of the system output at time t, $0 \le t \le T$, at the kth iteration etc. The notations e(t) , x(t) and u(t) will be similarly defined. In this algorithm the (k+1)th iteration control is obtained by manipulating the data stored at the kth ,(k-1)th iterations. Fig.1 shows the algorithm schematically. It is observed that , for fixed t, the algorithm in eq. (2) is a third – order difference equation of iteration number k.

3.MATHIMATICAL DESCRIBATION:

To show the convergence proof for eq.(2), let the linear time invariant system of eq.(1) be given with the initial state being known say $x(0) = x_o$. Let the axa matrices Ai , i=1,2 and axm matrices Bi , i=1,2 be given , so that the following conditions must be satisfied

$$\| A_1 - B_1 D \|_\infty + \| A_2 - B_2 D \|_\infty + \| A_3 - B_3 D \|_\infty < 1 \qquad \text{.....(3)}$$

And $A_1 + A_2 + A_3 = 1$ ....(4)

Then , for a given desired output $y_d(t)$, $0 \le t \le T$ the iterative control law of eq. (2) guarantees that , for each $t \in [0,T]$,
$y_k(t) \to y_d(t)$ as $k \to \infty$, if $u_o(t)$ for each $0 \le t \le T$ is chosen to be continuous and $x_k(t) = x_o$ for all k=1,2,3,……

Proof:

Let $u_d(t)$ and $x_d(t)$ be a control input and a system state , respectively that yield the desired output $y_d(t)$. It then follows from eqs. 1,2 and 4 that

$$u_d(t) - u_{k+1}(t) = A_1[u_d(t) - u_k(t)] + A_2[u_d(t) - u_{k-1}(t)] + A_3[u_d(t) - u_{k-2}(t)] - B_1[Cx_d(t) + D u_d(t) - Cx_k(t) - D u_k(t)] - B_2[Cx_d(t) + D u_d(t) - Cx_{k-1}(t) - D u_{k-1}(t)] - B_3[Cx_d(t) + D u_d(t) - Cx_{k-2}(t) - D u_{k-2}(t)] \qquad \text{.......(5)}$$

$$u_d(t) - u_{k+1}(t) = [A_1 - B_1 D][u_d(t) - u_k(t)] + [A_2 - B_2 D][u_d(t) - u_{k-1}(t)] + [A_3 - B_3 D][u_d(t) - u_{k-2}(t)] - B_1 C[x_d(t) - x_k(t)] - B_2 C[x_d(t) - x_{k-1}(t)] - B_3 C[x_d(t) - x_{k-2}(t)] \qquad \text{.........(6)}$$

Computing the norms of both sides , we obtain:

$$\| u_d(t) - u_{k+1}(t) \|_\infty \le \| A_1 - B_1 D \|_\infty . \| u_d(t) - u_k(t) \|_\infty + \| A_2 - B_2 D \|_\infty . \| u_d(t) - u_{k-1}(t) \|_\infty$$

$$+ \| A_3 - B_3 D \|_\infty . \| u_d(t) - u_{k-2}(t) \|_\infty + \| B_1 C \|_\infty . \| x_d(t) - x_k(t) \|_\infty + \| B_2 C \|_\infty .$$

$$\| x_d(t) - x_{k-1}(t) \|_\infty + \| B_3 C \|_\infty . \| x_d(t) - x_{k-2}(t) \|_\infty \quad \text{for all } t \in [0,T] \qquad \dots\dots\dots(7)$$

Now , because $x_d(0) = x_k(0)$ for all k, we obtain for $t \in [0,T]$

$$\| x_d(t) - x_k(t) \|_\infty = \left\| \int_0^t \{ [A\, x_d(\tau) + B\, u_d(\tau)] - [A\, x_k(\tau) + B\, u_k(\tau)] \} d\tau \right\|_\infty$$

$$\le \int_0^t \{ \| A \|_\infty . \| x_d(\tau) - x_k(\tau) \|_\infty + \| B \| . \| u_d(\tau) - u_k(\tau) \|_\infty \} d\tau \quad \dots..(8)$$

Applying the Bellman – Gronwall lemma [8,9], we obtain the following for $t \in [0,T]$

$$\| x_d(t) - x_k(t) \|_\infty \le \int_0^t \| B \|_\infty . \| u_d(t) - u_k(t) \|_\infty . e^{a(t-\tau)} d\tau \qquad \dots.(9)$$

Therefore combining Eqs. (8) and (9) we see that with

$$\partial u_k(t) = u_d(t) - u_k(t)$$

$$\| \partial u_{k+1}(t) \|_\infty \le R_1 \| \partial u_k(t) \|_\infty + R_2 \| \partial u_{k-1}(t) \|_\infty + R_3 \| \partial u_{k-2}(t) \|_\infty$$

$$+ S_1 \int_0^t \| \partial u_k(\tau) \|_\infty . e^{a(t-\tau)} d\tau + S_2 \int_0^t \| \partial u_{k-1}(\tau) \|_\infty . e^{a(t-\tau)} d\tau$$

$$+ S_3 \int_0^t \| \partial u_{k-2}(\tau) \|_\infty . e^{a(t-\tau)} d\tau \qquad \dots\dots(10)$$

where

$R_1 = \| A_1 - B_1 D \|_\infty$, $S_1 = \| B_1 C \|_\infty . \| B \|_\infty$

$R_2 = \| A_2 - B_2 D \|_\infty$, $S_2 = \| B_2 C \|_\infty . \| B \|_\infty$

$R_3 = \| A_3 - B_3 D \|_\infty$, $S_3 = \| B_3 C \|_\infty . \| B \|_\infty$

For all $t \in [0,T]$

Multiplying Eq.(10) by the positive function e^{-mt} we have

$$e^{-mt}\|\partial u_{k+1}(t)\|_\infty \le L_1 e^{-mt}\|\partial u_k(t)\|_\infty + L_2 e^{-mt}\|\partial u_{k-1}(t)\|_\infty + L_3 e^{-mt}\|\partial u_{k-2}(t)\|_\infty + m_1\int_o^t \|\partial u_k(\tau)\|_\infty e^{(a-m)(t-\tau)}d\tau + m_2\int_o^t \|\partial u_{k-1}(\tau)\|_\infty e^{(a-m)(t-\tau)}d\tau + m_3\int_o^t \|\partial u_{k-2}(\tau)\|_\infty e^{(a-m)(t-\tau)}d\tau \quad \ldots\ldots(11)$$

Therefore

$$\|\partial u_{k+1}(t)\|_m \le \left[R1 + \frac{S_1}{(m-a)}(1-e^{(a-m)T})\right]\|\partial u_k(t)\|_m \left[R2 + \frac{S_2}{(m-a)}(1-e^{(a-m)T})\right]\|\partial u_{k-1}(t)\|_m \left[R3 + \frac{S_3}{(m-a)}(1-e^{(a-m)T})\right]\|\partial u_{k-2}(t)\|_m \ldots(12)$$

Then $\|\partial u_k(t)\| \to 0$ as $k\to\infty$ if

$$\left[R_1 + \frac{S_1}{(m-a)}(1-e^{(a-m)T})\right] + \left[R_2 + \frac{S_2}{(m-a)}(1-e^{(a-m)T})\right] + \left[R_3 + \frac{S_3}{(m-a)}(1-e^{(a-m)T})\right] < 1 \ldots(13)$$

Noting that the nonnegative sequence $\{w_k\}$, k=1,2,3,…..
With the property

$$w_{k+1} \le r\, w_k + s\, w_{k-1} + n\, w_{k-2} \qquad (r,s,n>0) \qquad \ldots.(14)$$

converges to zero , r+s+n <1 holds, and observing from the property of eq.(14) that:

$$R_1+R_2+R_3<1 \qquad \ldots\ldots.(15)$$

We can choose m>0 large enough so that

$$R1+R2+R3+\frac{S_1}{(m-a)}(1-e^{(a-m)T})+\frac{S_2}{(m-a)}(1-e^{(a-m)T})+\frac{S_3}{(m-a)}(1-e^{(a-m)T}) \qquad \ldots..(16)$$

Thus $\|\partial u_k(.)\|_m \to 0$ as $k\to\infty$, by the definition of $\|.\|_m$ we know that

$$\sup_{t\epsilon[0,T]}\|\partial u_k(t)\|_\infty \le e^{mt}\|\partial u_k(t)\|_m \qquad \ldots\ldots(17)$$

therefore

$\sup_{t\epsilon[0,T]}\|\partial u_k(t)\|_\infty \to 0$ as $k\to\infty$ which means that $u_k(t)\to u_d(t)$ as $k\to\infty$, on t ϵ[0,T], furthermore , it implies that $x_k(t) \to x_d(t)$ as $k\to\infty$ on t ϵ[0,T]
Thus from above we conclude that $y_k(t)\to y_d(t)$ as $k\to\infty$ on t ϵ[0,T].

Furthermore , if there is no direct coupling between input and output , D matrix is nonsingular then the conditions in eq. (3) and (4) is not satisfied and the algorithm in eq. (2) is failed . The proposed algorithm in this case takes the following form

$u_{k+1}(t) = A_1 u_k(t) + A_2 u_{k-1}(t) + A_3 u_{k-2}(t) + B_1 \dot{e}_k(t) + B_2 \dot{e}_{k-1}(t) + B_3 \dot{e}_{k-2}(t)$…(18)

where $\dot{e}_k(t) = \dot{y}_d(t) - \dot{y}_k(t)$

And the conditions for convergence become

$\|A_1 - B_1CB\|_\infty + \|A_2 - B_2CB\|_\infty + \|A_3 - B_3CB\|_\infty < 1$ …..(19)

And $A_1 + A_2 + A_3 = 1$ ….(20)

In this case if CB=0 then the conditions for convergence is failed. The proposed algorithm takes the form:

$u_{k+1}(t) = A_1 u_k(t) + A_2 u_{k-1}(t) + A_3 u_{k-2}(t) + B_1 \ddot{e}_k(t) + B_2 \ddot{e}_{k-1}(t) + B_3 \ddot{e}_{k-2}(t)$…(21)

where $\ddot{e}_k(t) = \ddot{y}_d(t) - \ddot{y}_k(t)$

And the conditions for convergence become

$\|A_1 - B_1CAB\|_\infty + \|A_2 - B_2CAB\|_\infty + \|A_3 - B_3CAB\|_\infty < 1$ …..(22)

And $A_1 + A_2 + A_3 = 1$ ……(23)

The convergence proof for the algorithms in (18) and (21) is similar to the case of the algorithm in (2).

4. ***EXAMPLES***

In the following , we shall consider three examples. In these examples, we take the three different types recursive formulas described in eq.(2),(18) and (21). In each case , the performance of the proposed algorithm is compared with that of the method in reference [9].

EXAMPLE 1

Consider the following system

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = [-1 \quad -2] \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + u(t) \qquad \text{….(24)}$$

and let $y_d(t) = 12\,t(2-3\,t)$ for $0 \le t \le 1$ sec

In this case there is a direct coupling between input and output, D=1 so, the proposed algorithm of eq.(2) is considered in this case . Then we can obtain $e_k(t)$ for k_{th} iteration as shown in fig: 2. Here the control gains in eq.(2) are chosen as

$A_1 = 1.075$, $A_2 = -0.1$, $A_3 = 0.025$, $B_1 = 1.3$, $B_2 = -0.15$, $B_3 = -0.1$.

We need three iterations to converge to the desired tracking. From this figure , we can observe that four iterative trails are sufficient to generate a trajectory within the error bound of the desired trajectory $y_d(t)$. Our proposed learning control method is faster than the learning control method proposed in [9] which needs five iterations to track the desired trajectory.

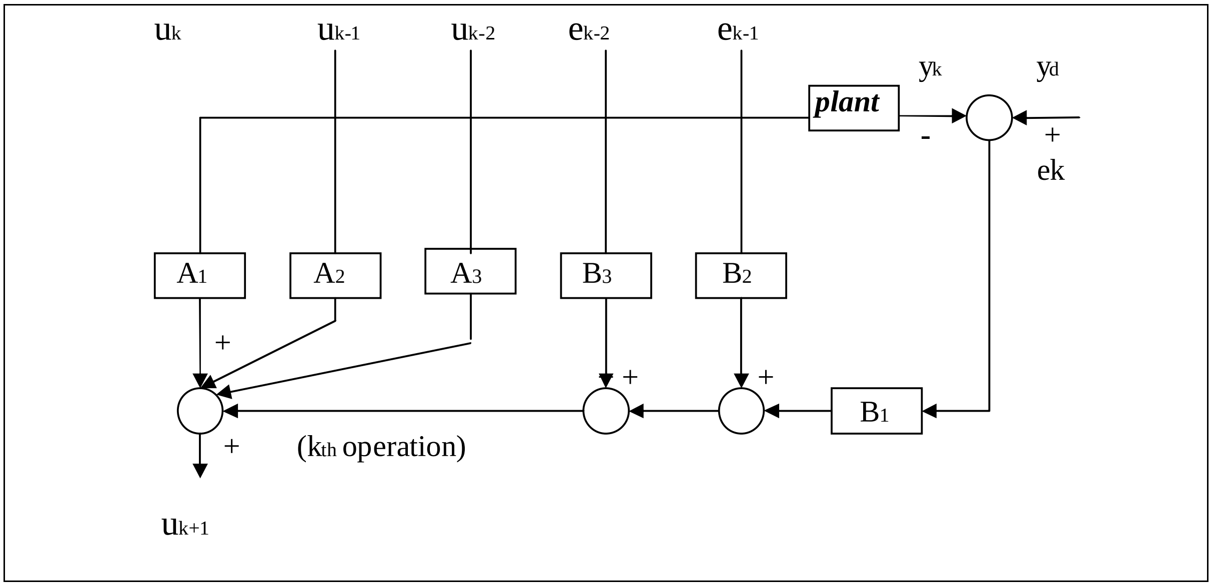

Fig :1. Structure of third order learning control scheme.

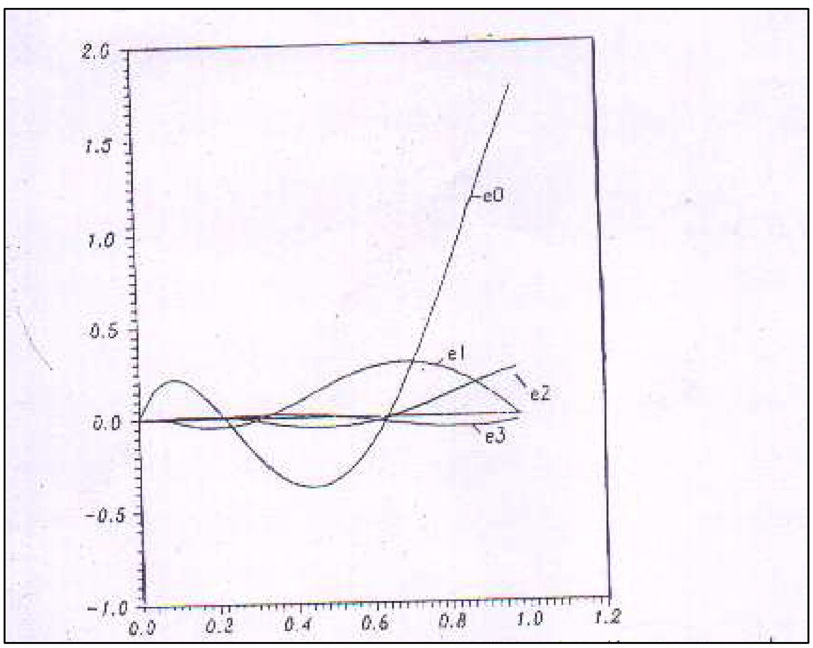

Fig: 2. Error tracking performance by the proposed method(for example 1).

EXAMPLE 2

Consider the system as follows:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = [0 \quad 1]\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \qquad \text{…..(25)}$$

and let $y_d(t) = 12\, t^2 (1-t)$ for $0 \le t \le 1$ sec
Since D=0 , the proposed algorithm of eq. (18) is considered for this case. We need two iterations to get the desired trajectory which gives faster convergence compared with the method in [9]. Results are shown in fig.3.

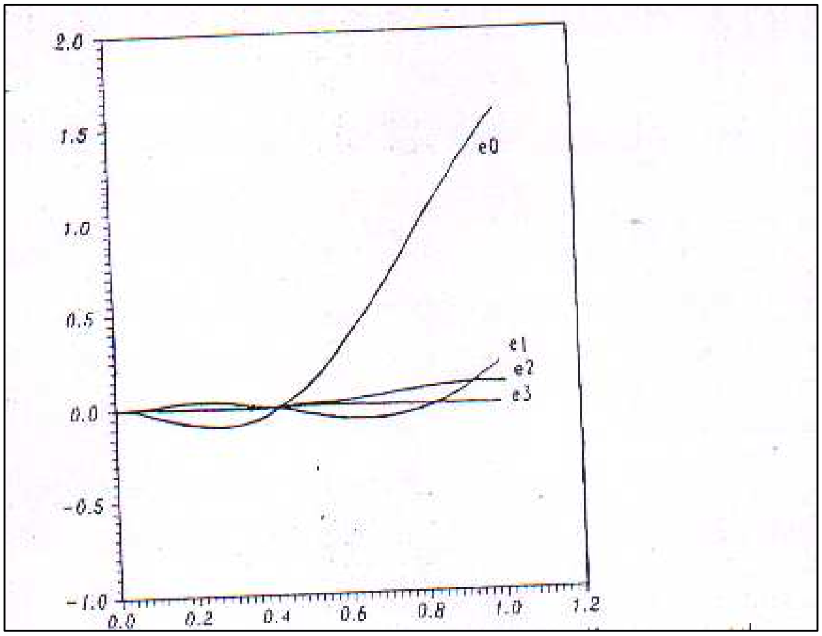

Fig: 3 . Error tracking performance by the proposed method(for example 2)

<u>Example3</u>
Consider the system as follows:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix}\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = [1 \quad 0]\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \qquad \text{…..(26)}$$

and let $y_d(t) = t^3 (4-3\,t)$ for $0 \le t \le 1$ sec.
Since D=0, CB=0, the proposed algorithm in eq. (21) is considered for this case. We need one iteration to get the desired trajectory , as shown in fig. 4 and it is faster compared with that in [9] which needs two iterations.

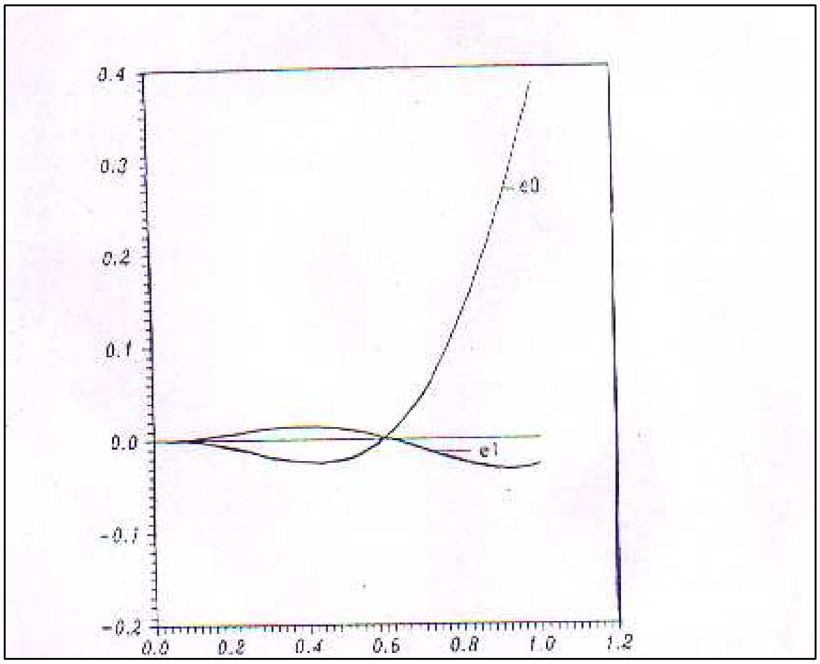

Fig.4 Error tracking performance by the proposed method(for example 3)

5. CONCLUSION

In this paper a general third order learning control method is proposed. Examples are given to show the effectiveness of the proposed algorithm. The convergence proof is given for linear system although it is applicable for nonlinear system. The proposed methodology turns out to be very simple and requires only a very rough description of the manipulator to be controllers but its main disadvantage is obviously represented by the necessity to perform different trails for any new trajectory which must be tracked. At the present moment , research activities are devoted to investigate about the possibilities of decomposing the problem of following a complex trajectory into a set of sub problems, each one referring to a simpler (and already learned) trajectory. The proposed higher-order iterative learning control algorithm uses more historical data but betters the output tracking performance.

6. REFERENCE

[1] S. Arimoto , S. Kawwamra, and F. Miyazaki, J. of Robotics systems, vol.1, no.2,pp. 123-140, march, 1984.

[2] S. Kawamura, F. Miyazaki and S. Arimoto, Proc. Of 24 th IEEE conf. on decision and control for Lauderdale, Fl,pp.1381-1386,1985.

[3] S. Arimoto and T. Naniwa ,Int. J. of Adaptive control and signal proc.,vol.4, no.5,pp. 453-564, July 1990.

[4] S. Arimoto , vol.74, no.2, pp. 235-243, Feb,1991.

[5] T. Naniwa, S. Arimoto , IEEE Trans. on robotics and Auto, vol.11, NO.3, PP.432-441, June 1995.

[6] J. J. Craig, Proc. American control conference, San Diego, pp.1566-1573,June,1984.

[7] P. Bondi , G. Gasalino and L. Gambardella, IEEE J. of robotics and Auto , vol.4 , no. 1, pp.14-22 , Feb , 1998.

[8] Z. Bien and K. M. H.OH, IEEE, Proc., vol.136,no.3,pp.105-112,May 1989.

[9] K. M. Omran," On the learning contol of robot manipulators",M.SC Thesis, Basrah University, Electrical Engineering Department, October ,199

الخلاصة :

ان التحكم المتعلم هي طريقة تحكم تستخدم لتحسين الاستجابة خلال فترة زمنية محددة بواسطة تكرار المحاولة . وهي طريقة تحكم تتابعية لاتحتاج الى حسابات كبيرة وتحتاج الى معرفة قليلة بالنظام المراد التحكم به ، ويمكن استخدامه كطريقة تحكم في الزمن الحقيقي للأنظمة المعقدة مثل معالجات الروبرت .في هذا البحث تم استخدام الخوارزمية لتطوير خصائص النظام في المحاولة التالية اعتمادا" على اكثر من قيمة سابقة للخطأ في المسارات التي يتم توليدها لكل محاولة . ثم اعطاء اثبات رياضي لصحة الطريقة معززا بالامثلة لتوضيح فعالية الطريقة المقترحة .