Pseudo-Inverse Based Iterative Learning Control for Linear Nonminimum Phase Plants with Unmodeled Dynamics*

Jayati Ghosh
Agilent Technologies Inc., Bio-Research Solutions, 5301 Stevens Creek Blvd. Santa Clara, CA 95051.
e-mail: jayati_ghosh@agilent.com

Brad Paden
Department of Mechanical and Environmental Engineering, University of California, Santa Barbara, CA 93106, U.S.A.
e-mail: paden@engineering.ucsb.edu

Learning control is a very effective approach for tracking repetitive processes. In this paper, the stable-inversion based learning controller as presented in (Ghosh, J. and Paden, B., 1999, "Iterative Learning Control for Nonlinear Nonminimum Phase Plants with Input Disturbances," in Proc. of American Control Conference; Ghosh, J. and Paden, B., 1999, "A pseudo-inverse based Iterative Learning Control for Nonlinear Plants with Disturbances," in Proc. of 38th Conference on Decision and Control.) is modified to accommodate linear nonminimum phase plants with uncertainties. The design of the learning controller is based on the computation of an approximate inverse of the nominal model of the linear plant, rather than its exact inverse. The advantages of this approach are that the output of the plant need not be differentiated and also the plant model need not be exact. A low pass zero-phase filter is used in the iteration loop to achieve robustness to plant uncertainty. The structure of the controller is such that the low frequency components of the trajectory converge faster than the high frequency components. [DOI: 10.1115/1.1789540]

1 Introduction

Iterative learning control (ILC) refers to a class of self-tuning controllers where the system performance of a specified task is gradually improved or perfected based on the previous performances of identical tasks. The concept of "practice" to improve performance is emphasized in all areas of human endeavor such as gymnastics or musical performances. In the same way, learning controllers attempt to tune the performance of a system on a given trajectory by learning through practice. Additionally, learning control can be used effectively when the plant cannot be modeled accurately. The most commonly seen applications of learning control are in the area of robot control in manufacturing and production industries. Typically, a robot is required to perform a single task, say pick-and-place an object along a given trajectory, repetitively. With a feedback controller alone, the same tracking error would persist in every repeated trial. In contrast, a learning controller can use the information from the previous executions to improve the tracking performance in the next execution. While in some applications, the need to repeat a trajectory multiple times for learning may be a distinct disadvantage, in many other applications, repetitive tasks are commonly performed making learning control a very natural solution. For more than a decade researchers have defined and analyzed iterative learning control (ILC) schemes. First introduced in 1984 by Arimoto et al. [1] and Craig [2] and later modified by many others including Kawamura et al. [3], Atkeson et al. [4], and Bondi et al. [5], ILC schemes strive to improve the performance of repetitive tasks using the information of the previous trial of the same task. Modifications of the basic ILC algorithm, such as *p*-type, PD-type and PID-type have evolved in the process. The robustness of the ILC algorithms to disturbances, uncertainties, and initialization errors is still an active area of research. Arimoto [6] analyzed the robustness of a PI-type ILC algorithm to errors in initialization, measurement and fluctuation during operation, and introduced a forgetting factor into the ILC to enhance it. It is shown in [7] that the model-based learning scheme in [4] is a special case of this more general approach. Sugie and Ono [8] demonstrate the necessity of the use of the error derivative in the learning control process for plants that do not have a direct transmission term. Based on the time-varying nonlinear extension result of Hauser [7], Heinzinger et al. [9] showed that under certain assumptions, the asymptotic tracking errors are bounded and the bounds are continuous functions of the initial errors, uncertainties and disturbances. Chen et al. [10] proposed a way to adjust the final tracking error bound to a prescribed level with the use of the current iteration tracking error, in the presence of uncertainties, disturbances, and initialization errors. Gao and Chen [11] have illustrated with counter examples the limitations of some of these learning algorithms with regard to nonminimum phase systems. To remove the minimum phase requirement, they developed a new learning algorithm for linear systems based on "stable inversion." Based on the algorithm of Gao and Chen, we developed an iterative learning control algorithm in [12] for nonlinear nonminimum phase plants with input disturbances. A modification of the iterative learning control algorithm presented in [12] is presented in [13] so that it can be applied to a more generic class of nonlinear nonminimum phase plants with input disturbance and output sensor noise.

In this paper we propose an extension to the iterative learning control algorithm presented in [12,13] so that it can be applied to a nonminimum phase plant with neglected and unmodeled dynamics. Either through deliberate neglect or because of the lack of understanding or the lack of proper measurement of the physical process. Usually, at high frequencies, any model of a real system will contain some unmodeled dynamics. In Sec. 2, an update law for the learning control is proposed for a linear plant with uncertainty. A learning controller (Fig. 1) based on a pseudo-inverse of the nominal model of the plant is applied as proposed in [13]. A proof of convergence of the input trajectory to a neighborhood of the desired one is provided. In Sec. 3, simulation examples are presented to show the performance of the proposed learning controller. Finally, Sec. 4 concludes the paper.

2 Linear Nonminimum Phase Plant With Model Uncertainty

In this section we present a robust iterative algorithm for SISO stable linear plants with model uncertainty. Let H be a nominal model of a stable SISO linear plant.

In the state-space form H is given as:

$$\dot{x}(t)=Ax(t)+Bu(t), \quad x(\pm\infty)=0$$

$$y(t)=Cx(t). \tag{1}$$

$$H{:}u\mapsto y; \quad L_\infty[0,\infty)\to L_\infty[0,\infty).$$

A desired trajectory $y_d(t)$ is supported on a finite interval ($t\in[0,T]$) of the time axis. The objective of learning is to construct a sequence of input trajectories $\{u_i\}_{i=1}^{\infty}$ such that $u_i\to u_d$ and $u_d(t)$ causes the system to track a trajectory $y_d(t)$ as closely as possible on $[0,T]$, i.e.:

---

*This research was supported by the National Science Foundation under Grant No. CMS-9800294.

Contributed by the Dynamic Systems, Measurement, and Control Division of THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS for publication in the ASME JOURNAL OF DYNAMIC SYSTEMS, MEASUREMENT, AND CONTROL. Manuscript received by the ASME Dynamic Systems and Control Division December 26, 2003. Associate Editor: F. Ghorbel.

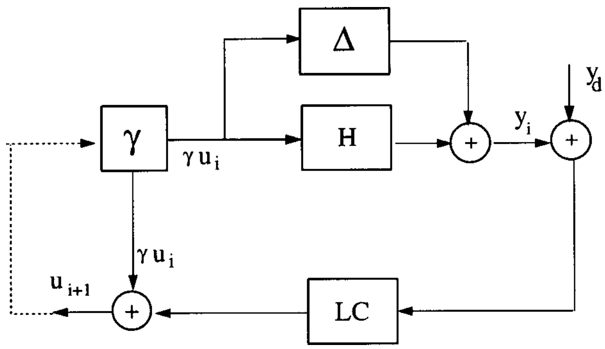

Fig. 1 Learning control system: (H) nominal plant model; (Δ) unmodeled dynamics, (γ) zero-phase filter; (LC) learning controller

$y_d=(H+\Delta)(u_d)$, where Δ represents unmodeled dynamics. Consider the I–O map for the adjoint system H^*:

$$\dot{\bar{x}}=-A^T\bar{x}-C^T\bar{u}, \quad \bar{x}(\pm\infty)=0$$
$$\bar{y}=B^T\bar{x}. \tag{2}$$

Since A *is Hurwitz*, $-A^T$ *is hyperbolic and has all eigenvalues on the right-half plane. Furthermore, Eq.* (2) *defines a unique noncausal mapping as shown by Devasia et al.* [14]:

$$H^*:\bar{u}\mapsto\bar{y}; \quad L_\infty\to L_\infty.$$

The adjoint system satisfies the property $\langle Hu,v\rangle=\langle u,H^*v\rangle$ [15].

Defining $H^{\dagger,\alpha}$

Motivated by the concept of a pseudo-inverse [16,17] we define learning controller by the following linear operator:

$$H^{\dagger,\alpha}\triangleq(\alpha I+H^*H)^{-1}H^* \tag{3}$$

for $\alpha\neq 0$. We call this "approximate inverse" the α-pseudo inverse of H. For simplicity the α-pseudo-inverse is referred to as simply a pseudo-inverse in the rest of this paper.

In time-domain $(\alpha I+H^*H)$: $\delta u\to\delta\tilde{y}$ is:

$$\begin{bmatrix}\delta\dot{x}\\ \delta\dot{\bar{x}}\end{bmatrix}=\begin{bmatrix}A & 0\\ -C^TC & -A^T\end{bmatrix}\begin{bmatrix}\delta x\\ \delta\bar{x}\end{bmatrix}+\begin{bmatrix}B\\ 0\end{bmatrix}\delta u; \quad \begin{bmatrix}\delta x(\pm\infty)\\ \delta\bar{x}(\pm\infty)\end{bmatrix}=0$$
$$\delta\tilde{y}=\alpha\delta u+\delta\bar{y}.=\alpha\delta u+B^T\delta\bar{x} \tag{4}$$

Since H is stable, the above system Eq. (4) is hyperbolic with eigenvalues $\lambda(A)\cup\lambda(-A^T)$. Hence, by [14] $(\alpha I+H^*H)$: $\delta u\mapsto\delta\tilde{y}$; $L_\infty\to L_\infty$ and is noncausal. Solving Eq. (4) for δu, we see that the inverse operator $(\alpha I+H^*H)^{-1}$ is:

$$\begin{bmatrix}\delta\dot{x}\\ \delta\dot{\bar{x}}\end{bmatrix} = \underbrace{\begin{bmatrix}A & -B\alpha^{-1}B^T\\ -C^TC & -A^T\end{bmatrix}}_{A_\alpha}\begin{bmatrix}\delta x\\ \delta\bar{x}\end{bmatrix}+\begin{bmatrix}B\alpha^{-1}\\ 0\end{bmatrix}\delta\tilde{y}; \quad \begin{bmatrix}\delta x(\pm\infty)\\ \delta\bar{x}(\pm\infty)\end{bmatrix}=0$$
$$\delta u = \alpha^{-1}(\delta\tilde{y}-B^T\delta\bar{x}) \tag{5}$$

The eigenvalues of the above system are continuous functions of α. In the limit $\alpha\to\infty$, A_α is hyperbolic (since A is Hurwitz). Thus we can always choose an α for which A_α is hyperbolic. The system Eq. (5) is solved by the stable-noncausal-solution approach of Devasia et al. [14] under the boundary conditions $X(\pm\infty)=0$. Hence, $(\alpha I+H^*H)^{-1}$: $\delta\tilde{y}\mapsto\delta u$; $L_\infty\to L_\infty$.

The learning controller $(\alpha I+H^*H)^{-1}H^*$ is given in time domain by:

$$\underbrace{\begin{bmatrix}\delta\dot{x}\\ \delta\dot{\bar{x}}\\ \delta\dot{z}\end{bmatrix}}_{X} = \underbrace{\begin{bmatrix}A & -B\alpha^{-1}B^T & B\alpha^{-1}B^T\\ -C^TC & -A^T & 0\\ 0 & 0 & -A^T\end{bmatrix}}_{A_c}\begin{bmatrix}\delta x\\ \delta\bar{x}\\ \delta z\end{bmatrix}+\begin{bmatrix}0\\ 0\\ -C^T\end{bmatrix}\delta y; \quad X(\pm\infty)=0$$
$$\delta u = \begin{bmatrix}0 & -\alpha^{-1}B^T & \alpha^{-1}B^T\end{bmatrix}\begin{bmatrix}\delta x\\ \delta\bar{x}\\ \delta z\end{bmatrix} \tag{6}$$

A_c is block diagonal, therefore the eigenvalues of A_c are eigenvalues of A_α [Eq. (5)] and $-A^T$. Since A_α is hyperbolic for some α, A_c is hyperbolic. Hence, the solution of the linear controller described by Eq. (12) can be obtained using stable-noncausal-solution approach [14] under the boundary conditions $X(\pm\infty)=0$. Hence,

$$(\alpha I+H^*H)^{-1}H^*:\delta y\mapsto\delta u$$
$$L_\infty\to L_\infty.$$

Using initial conditions at $t=-\infty$ rather than $t=0$ allows us to control $x(0)$ via δy over the interval $(-\infty,0]$. Thus tracking performance can be improved relative to assumptions of $u\equiv 0$ on $(-\infty,0$ and $x(0)=0$. Now the update law of the ILC is written in terms of the operators H, $(\alpha I+H^*H)^{-1}H^*$ and a low pass zero-phase filter γ as follows:

$$u_{i+1}(t)=\gamma u_i(t)+\delta u_i(t)=\gamma u_i(t)+(\alpha I+H^*H)^{-1}H^*[y_d(t)-y_i(t)], \tag{7}$$

where i is the index of iteration of ILC. Let $H(\omega):=\mathcal{F}(H)$, $\mathcal{F}(H^{\dagger,\alpha})=:[\alpha+H^*(\omega)H(\omega)]^{-1}H^*(\omega)$ and $\mathcal{F}(\gamma)=:\gamma(\omega)$ where $\mathcal{F}(A)$ stands for the Fourier transform of the operator A. We can write the update law in frequency domain as:

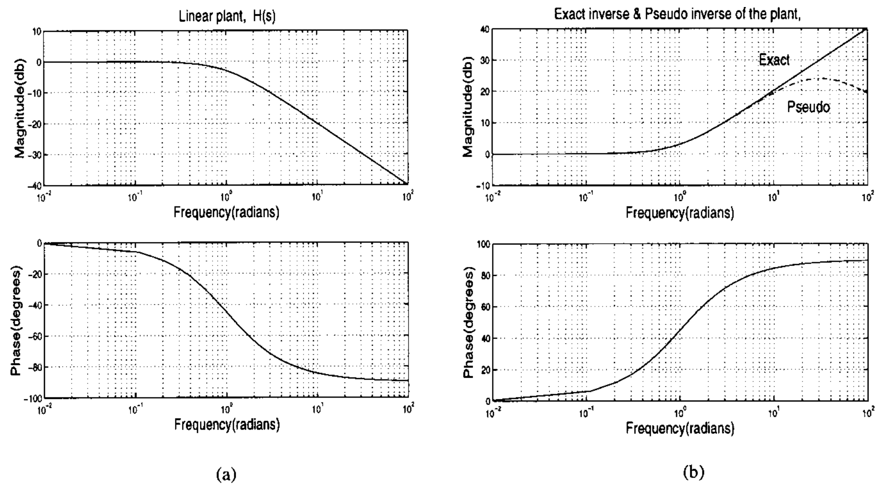

Fig. 2 Frequency responses (a) linear model H, (b) exact inverse H^{-1} and pseudo-inverse $H^{\dagger,\alpha}$

$$u_{i+1}(\omega)=\gamma(\omega)u_i(\omega)+[\alpha+H^*(\omega)H(\omega)]^{-1}H^*(\omega)[y_d(\omega)-y_i(\omega)]. \tag{8}$$

2.1 Convergence Analysis. In this section we present a theorem for the convergence of the input sequence to the neighborhood of the desired input solution u_d.

Condition 1:

$$|H^*(\omega)\Delta(\omega)\gamma(\omega)|<|H^*(\omega)H(\omega)|.$$

Theorem 1: Given a SISO stable linear plant with uncertainty, if Condition 1 is satisfied, the update law Eq. (8) produces a sequence of inputs which converges to a neighborhood of the desired input solution u_d if there are no disturbances and no initialization error.

Proof: In frequency domain the update law Eq. (8) can be written for any particular frequency ω as follows:

$$\begin{aligned}u_{i+1}(\omega)&=\gamma(\omega)u_i(\omega)+[\alpha+H^*(\omega)H(\omega)]^{-1}H^*(\omega)[y_d(\omega)-y_i(\omega)]\\&=\gamma(\omega)u_i(\omega)+[\alpha+H^*(\omega)H(\omega)]^{-1}H^*(\omega)\{Hy_d(\omega)-[H(\omega)+\Delta(\omega)]\gamma(\omega)u_i(\omega)\}=\gamma(\omega)u_i(\omega)+[\alpha\\&\quad+H^*(\omega)H(\omega)]^{-1}[H^*(\omega)Hy_d(\omega)-[H^*(\omega)H(\omega)+H^*(\omega)\Delta(\omega)+\alpha-\alpha)\gamma(\omega)u_i(\omega)]\\&=\gamma(\omega)u_i(\omega)-[\alpha+H^*(\omega)H(\omega)]^{-1}[\alpha+H^*(\omega)H(\omega)]\gamma(\omega)u_i(\omega)\\&\quad+[\alpha+H^*(\omega)H(\omega)]^{-1}[\alpha-H^*(\omega)\Delta(\omega)]\gamma(\omega)u_i(\omega)+[\alpha+H^*(\omega)H(\omega)]^{-1}H^*(\omega)y_d(\omega)\\&=[\alpha+H^*(\omega)H(\omega)]^{-1}[\alpha-H^*(\omega)\Delta(\omega)]\gamma(\omega)u_i(\omega)+[\alpha+H^*(\omega)H(\omega)]^{-1}H^*(\omega)(H(\omega)+\Delta)u_d(\omega).\end{aligned} \tag{9}$$

If Condition 1 is satisfied then from Eq. (9) we can show in the following way that $|[\alpha+H^*(\omega)H(\omega)]^{-1}[\alpha-H^*(\omega)\Delta(\omega)]\gamma(\omega)|<1$ and hence u_i converges pointwise in the frequency domain. The argument is as follows:

$$\begin{aligned}\text{Condition 1}&\Rightarrow|H^*(\omega)\Delta(\omega)\gamma(\omega)|<|H^*(\omega)H(\omega)|\\&<\alpha(1-|\gamma(\omega)|)+|H^*(\omega)H(\omega)|\\&(\text{Since, } |\gamma(\omega)|<1,\ \alpha>0)\\&\Rightarrow\alpha|\gamma(\omega)|+|H^*(\omega)\Delta(\omega)\gamma(\omega)|<|\alpha\\&\quad+H^*(\omega)H(\omega)|\quad\text{Since,}\quad H^*(\omega)H(\omega)>0\\&\Rightarrow|[\alpha-H^*(\omega)\Delta(\omega)]\gamma(\omega)|<\alpha+H^*(\omega)H(\omega)|\\&\Rightarrow|[\alpha+H^*(\omega)H(\omega)]^{-1}\\&\times\quad[\alpha-H^*(\omega)\Delta(\omega)]\gamma(\omega)|<1.\end{aligned} \tag{10}$$

2.2 Discussion. This ILC scheme has some advantages over that presented in Refs. [12], [13], though in this paper only application to linear plant is addressed. In [12] the inverse of the plant, H^{-1} is considered to be the learning operator. This necessitates taking the derivative of the output to invert the system. Sugie and Ono [8] demonstrated the need for differentiation in the learning operator of most of the existing ILC schemes for systems without direct feedthrough terms in output equations of plants. In practice, derivatives cannot be reliably computed in the presence of output sensor noise. Furthermore, the plant may itself produce an output signal that is not differentiable. In the learning algorithm based on a pseudo-inverse [13] however, it is not necessary to take the derivative of the output in order to calculate the update term of the system input at every iteration. (Note that α should be nonzero).

The frequency responses of the linearized plant H [as given by Eq. (1)] and its exact inverse H^{-1} and pseudo-inverse $H^{\dagger,\alpha}$ (with α=0.001) are shown in Figs. 2(a) and 2(b). In the ILC scheme presented in [12], the learning operator H^{-1} has high gain at high frequency as shown in Fig. 2(b). Therefore, the high frequency noise is amplified by the learning operator. The frequency response of the learning controller $H^{\dagger,\alpha}$ proposed in [13], is shown in Fig. 2(b) (with α=0.001). From Fig. 2(b) we see that the frequency response of $H^{\dagger,\alpha}$ behaves similarly to H^{-1} at low frequencies, but rolls off at high frequencies demonstrating a lowpass nature. Thus the high frequency sensor noise is filtered out. The phase responses of the exact inverse and the pseudo-inverse are identical [see Fig. 2(b)]. Note that $(\alpha+H^*H)$ is a *zero-phase filter*. Excellent tracking of the low frequency components is achieved after a few iterations, while the high frequency components of the output error signal converge more slowly but in order to design the controller an exact model of the linearized plant is used in [12,13]. In this paper we demonstrate that for the linear plant if $|\Delta(\omega)|<|H(\omega)|$, $\forall\omega$, the learning algorithm as presented in [13] will still converge. Otherwise, we need to design a zero-phase filter $\gamma(\omega)$ such that Condition 1 is met to achieve convergence.

3 Simulation Results

3.1 Simulation Results With Additive Uncertainty. In this section we perform simulation studies with a SISO stable plant whose nominal model is given by $H(s)=1/s+1$ and additive unmodeled dynamics $\Delta(s)=5/(s+25)$. The reference output trajectory is given by:

$$y_d(t)=\begin{cases}0.2\sin(t) & t\in[0,2\pi]\\ 0 & \text{otherwise}\end{cases}. \tag{11}$$

The learning controller $H^{\dagger,\alpha}$ is given by Eq. (12) with α=0.001, $A=-1$, $B=1$, and $C=1$, as:

$$\begin{bmatrix}\delta\dot{x}\\ \delta\dot{\bar{x}}\\ \delta\dot{z}\end{bmatrix} = \underbrace{\begin{bmatrix}A & -B\alpha^{-1}B^T & B\alpha^{-1}B^T\\ -C^TC & -A^T & 0\\ 0 & 0 & -A^T\end{bmatrix}}_{A_c}\begin{bmatrix}\delta x\\ \delta\bar{x}\\ \delta z\end{bmatrix}+\begin{bmatrix}0\\ 0\\ -C^T\end{bmatrix}\delta y;\quad X(\pm\infty)=0$$

$$\delta u=[0\quad -\alpha^{-1}B^T\quad \alpha^{-1}B^T]\begin{bmatrix}\delta x\\ \delta\bar{x}\\ \delta z\end{bmatrix}. \tag{12}$$

Eigenvalues of A_c are −31.6386, 31.6386, 1.0. Since the linear controller is unstable, we apply noncausal stable inversion [14].

Since Condition 1 is not satisfied (see Fig. 3) at high frequencies, the high frequency error spikes are still present after five iterations, though the low frequency components converge to zero (Fig. 4).

Now we include a zero-phase filter γ as shown in Fig. 5. We consider a low pass filter $\gamma_f(s)=10/(s+10)$. After filtering through $\gamma_f(s)$ in the forward direction, the filtered sequence is then time reversed and run back through the filter. The output of the second filtering operation is then time reversed to produce the output of γ. The result has precisely zero phase distortion and magnitude modified by the square of the filter's magnitude response. The effective filter is given as $\gamma(s)=\gamma_f(s)\cdot\gamma_f(-s)$. Tracking performance for the high frequency components improves when the filter is added.

4 Conclusion

The learning algorithm presented in this paper guarantees learning, under quite general assumptions. Theoretical assertions are

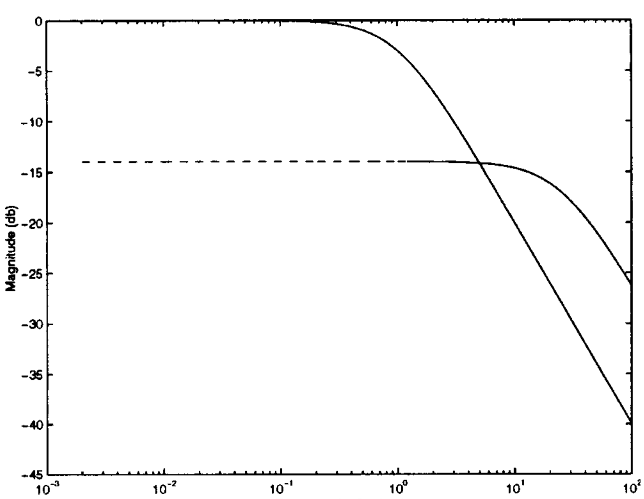

Fig. 3 Plot of the magnitude response of the nominal plant (solid line) and the plant uncertainty (dotted line)

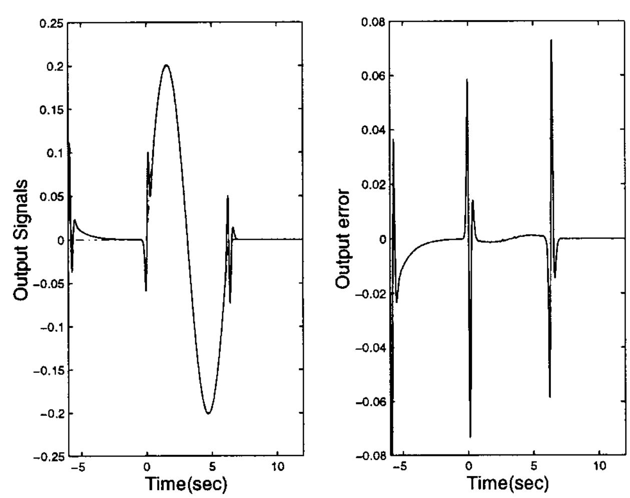

Fig. 4 Tracking with plant uncertainty and without filter $\gamma(s)$ after five iterations

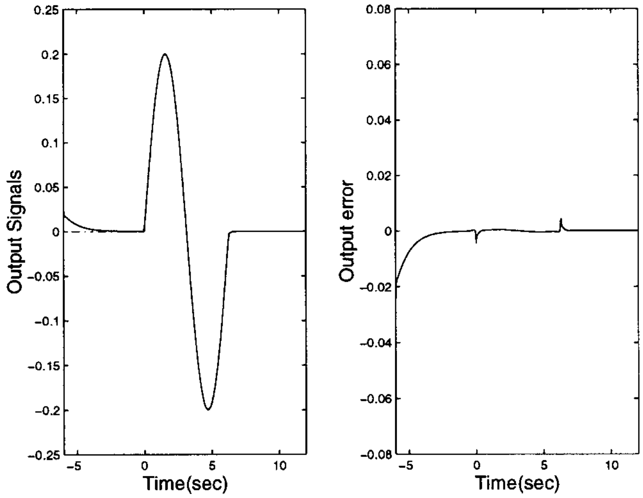

Fig. 5 Tracking with plant uncertainty and with filter $\gamma(s)$ after five iterations

corroborated by simulation results which demonstrate tracking in the presence of unmodeled dynamics. Extension of this algorithm to nonlinear plants is a topic of future research.

References

[1] Arimoto, S., Kawamura, S., and Miyazaki, F., 1984, "Bettering Operation of Dynamic Systems by Learning: A New Control Theory for Servomechanism or Mechatronics Systems," in *Proc. of 23rd Conference on Decision and Control*, pp. 1064–1069.

[2] Craig, J. J., 1984, "Adaptive Control of Manipulators Through Repeated Trials," in *Proceedings of American Control Conference*, pp. 1566–1573.

[3] Kawamura, S., Miyazaki, F., and Arimoto, S., 1988, "Realization of Robot Motion Based on a Learning Method," IEEE Trans. Syst. Man Cybern., **18**, pp. 126–133.

[4] Atkeson, C. G., and McIntyre, J., 1986, "Robot Trajectory Learning Through Practice," in *IEEE International Conference on Robotics and Automation*, pp. 1737–1742.

[5] Bondi, P., Casalino, G., and Gambardella, L., 1988, "On the Iterative Learning Control Theory for Robotic Manipulators," IEEE J. Rob. Autom., **4**, pp. 14–21.

[6] Arimoto, S., 1990, "Robustness of Learning Control for Robot Manipulators," in *IEEE International Conference on Robotics and Automation*, pp. 1528–1533.

[7] Hauser, J. E., 1987, "Learning Control for a Class of Nonlinear Systems," in *Proceedings of 28th Conference on Decision and Control*, pp. 859–860.

[8] Sugie, T., and Ono, T., 1991, "An Iterative Learning Control for Dynamical Systems," Automatica, **27**, pp. 729–732.

[9] Heinzinger, G., Fenwick, D., Paden, B., and Miyazaki, F., 1992, "Stability of Learning Control With Disturbances and Uncertain Initial Condition," IEEE Trans. Autom. Control, **AC-37**, pp. 110–114.

[10] Chen, Y., Wen, C., and Sun, M., 1997, "A Robust High-Order P-Type Iterative Learning Controller Using Current Iteration Tracking Error," Int. J. Control, **68**, pp. 331–342.

[11] Gao, J., and Chen, D., 1998, "Iterative Learning Control for Non-Minimum Phase Systems," in *Proc. of American Control Conference*.

[12] Ghosh, J., and Paden, B., 1999, "Iterative Learning Control for Nonlinear Nonminimum Phase Plants With Input Disturbances," in *Proceedings of American Control Conference*.

[13] Ghosh, J., and Paden, B., 1999, "A Pseudo-Inverse Based Iterative Learning Control for Nonlinear Plants With Disturbances," in *Proc. of 38th Conference on Decision and Control*.

[14] Devasia, S., Chen, D., and Paden, B., 1996, "Nonlinear Inversion-Based Output Tracking," IEEE Trans. Autom. Control, **AC-41**, pp. 930–942.

[15] Green, M., and Limebeer, D. J. N., 1995, *Linear Robust Control*. Prentice Hall Information and System Sciences Series, Prentice Hall.

[16] Luenberger, D. G., 1969, *Optimization by Vector Space Methods*. Wiley Interscience, New York.

[17] Desoer, C. A., and Whalen, B. H., 1963, "A Note on Pseudoinverses," J. Soc. Ind. Appl. Math., **11**, pp. 442–447.