Mechanical Systems and Signal Processing 23 (2009) 523–538

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/jnlabr/ymssp

B-spline network-based iterative learning control for trajectory tracking of a piezoelectric actuator

Fu-Shin Lee [a,*], Jhen-Cheng Wang [b], Chiang-Ju Chien [c]

[a] Department of Mechatronic Engineering, Huafan University, Taipei 223, Taiwan
[b] Department of Electronic Engineering, Tungnan University, Taipei 222, Taiwan
[c] Department of Electronic Engineering, Huafan University, Taipei 223, Taiwan

ARTICLE INFO

Article history:
Received 9 October 2006
Received in revised form
2 June 2008
Accepted 5 June 2008
Available online 11 June 2008

Keywords:
B-spline network
Iterative learning control
Piezoelectric actuator

ABSTRACT

This paper presents the trajectory tracking approach of a piezoelectric actuator using an iterative learning control (ILC) scheme based on B-spline network (BSN) filtering. The ILC scheme adopts a state-compensated iterative learning formula, which compensates for the state difference between two consecutive iterations in order that the iterative learning can learn from the tracking errors of the previous iteration effectively. The BSN is used to attenuate the noises and retrieve the signals of the tracking errors for the ILC. The BSN serves as a unique filter which generally does not have zero-phase responses. Design details on the ILC scheme using BSN filtering are discussed in the paper. Extensive experiments of tracking two desired trajectories for a piezoelectric actuator are presented. The experimental results show that the state-compensated ILC scheme using BSN filtering can achieve fast error convergence and keep small steady-state tracking errors close to the system noise level. This research thus relaxes the restriction of the zero-phase criterion commonly applied to the ILC filtering in the literature.

1. Introduction

Owing to the advantages of infinite small displacement, high stiffness, and wide bandwidth, piezoelectric actuators have been commonly used for precision positioning in various engineering fields, such as positioning of diamond machine tools, positioning of optical lenses, and positioning of masks in semiconductor manufacturing. However, the piezoelectric actuators are nonlinear devices, and can be characterized by certain hysteresis phenomena between their input voltages and output displacements. Various control schemes have been proposed in the literature to compensate for the nonlinear behaviors of the actuators. For example, there were schemes using the Preisach model-based control [1] and the neural sliding-mode control [2].

Differing from other endeavors of control complexities, some studies initiated the simpler approach using the iterative learning control (ILC) [3]. Certain degrees of success were achieved in tracking control of piezoelectric actuators by using the P-type ILC, the current-error-assisted D-type ILC, the model-assisted ILC, and the state-compensated ILC schemes [4–8]. The state-compensated ILC (SCILC) proposed by the authors [8] is a novel scheme, which enhances the traditional ILC by adding a state compensation enhancement to the iterative learning. In [8], a zero-phase (ZP) finite impulse response (FIR) filter was used to filter the learnable errors for the ILC. The major goal of this paper, on the contrary, is to relax the restriction of ZP filtering by adopting the B-spline network (BSN) for error filtering. The BSN consists of a number of

* Corresponding author. Tel.: +886 2 2663 2102; fax: +886 2 2663 3173.
E-mail addresses: fslee@huafan.hfu.edu.tw (F.-S. Lee), jcwang@mail.tnu.edu.tw (J.-C. Wang), cjc@huafan.hfu.edu.tw (C.-J. Chien).

0888-3270/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ymssp.2008.06.003

B-spline basis functions to realize an input–output mapping. The mapped output is a linear combination of these basis functions. The BSN structure has been proposed in the learning feed-forward control (LFFC) [9–15] to enhance the performance of feedback control systems. The BSN in the LFFC is tuned by the feedback control effort of the previous iteration to generate a feed-forward control effort in the current iteration. The feed-forward control effort is thus modified iteratively in order that the system output will approach the desired trajectory eventually. The BSN is basically a function approximator [15], and the tuning mechanism of the BSN is in essence a general non-ZP filter [10]. The BSN is adopted in this research to filter the learnable errors of the previous iteration without a zero phase in general. The filtered errors are then used for learning in the current iteration.

The remainder of the paper is organized as follows. Section 2 describes the experimental setup. Section 3 introduces the SCILC. In Section 4, the BSN filtering is discussed. In Section 5, a convergence analysis is performed for the SCILC. Section 6 addresses the time-frequency analysis of tracking errors. Section 7 presents the controller design details and the experimental results. Finally, Section 8 concludes this paper.

2. Experimental setup

The experimental platform in this research consists of a piezoelectric actuator unit, an electronics unit, a data acquisition device, and a personal computer, as shown by Fig. 1. The piezoelectric actuator has the displacement from −8 to 40 μm corresponding to the input voltage from −30 to +150 V. The actuator has a built-in strain gauge which is connected to the sensor amplifier of the electronics unit. The amplifier outputs a sensing voltage ranging from +0.2 to −1 V. The data acquisition device samples the sensing voltage for the control program and delivers the control effort to the power amplifier, which has a gain of 30 to drive the actuator. The data acquisition device has 12-bit A/D and D/A converters with a maximum sampling rate of 100 K-samples per s for input and 1 K-samples per s for output.

The control program is written in MATLAB, and it utilizes Data Acquisition Toolbox functions of MATLAB to perform online learning control within the Windows operating system. The program implements the SCILC scheme using the B-spline network filtering technique, as introduced in the following.

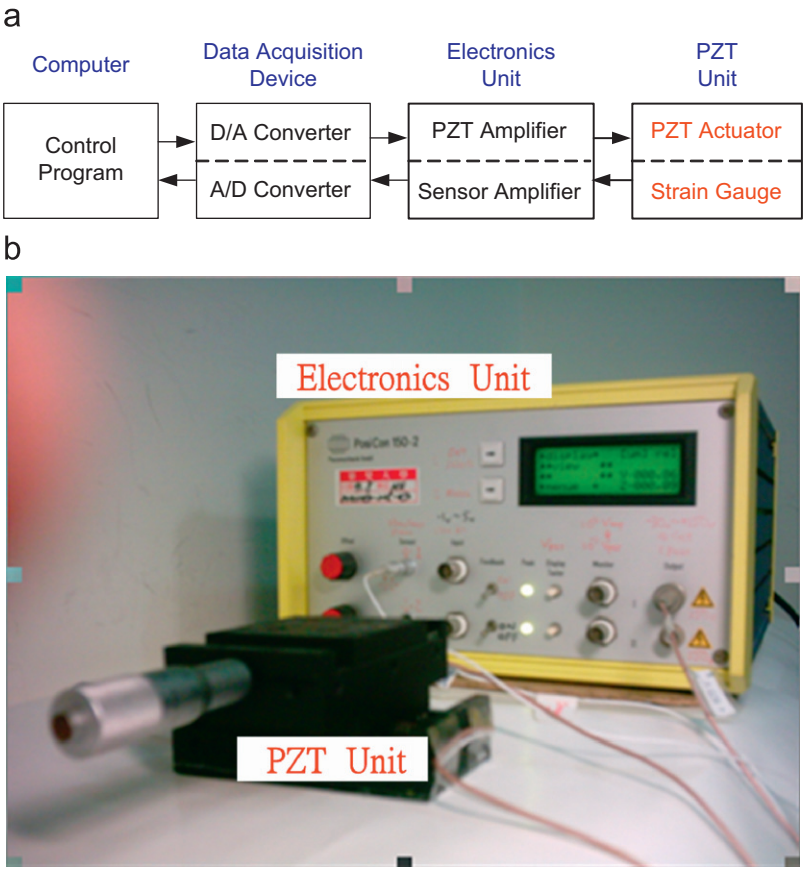

Fig. 1. (a) Block diagram of the experimental setup. (b) Picture of the electronics unit and the PZT unit.

3. State-compensated ILC

A variety of ILC formulas have been proposed in the literature. A notable class of iterative learning control is the current iteration tracking error ILC (CITE-ILC). It has a typical update law in the discrete-time domain [6]:

$$u_i(n) = u_{i-1}(n) + L\ e_{i-1}(n+1) + K\ e_i(n) \tag{1}$$

where u is the input, e is the output error, L is the learning gain, K is the proportional gain, n is the time index, and the subscripts i and $i-1$ denote the current iteration and the previous iteration, respectively. In theory, error convergence of Eq. (1) can be proved with certain assumptions such as the initial state restriction and the Lipschitz condition [16].

To relax the restrictions on traditional ILC such as the CITE-ILC, the authors proposed the SCILC [8] with the formula

$$u_i(n) = u_{i-1}(n) + L\ e_{i-1}(n+1) + K[e_i(n) - e_{i-1}(n)] \tag{2}$$

In Eq. (2), the difference between $e_i(n)$ and $e_{i-1}(n)$ means the difference of output error at time instant n between iteration i and iteration $i-1$. This difference of output error is the same in magnitude as the difference of system output. It indeed implies the difference of system state at time instant n between the two iterations, when the system output is used as one major state variable. Therefore, the term $K[e_i(n)-e_{i-1}(n)]$ can offset the state difference between the current and previous iterations at any time instant n. Using the state compensation, the SCILC can relax the convergence assumptions for traditional ILC and yield more precise tracking [8].

In Eq. (2), the learnable error is $e_{i-1}(n+1)$, which is the error happening in the previous iteration. This tracking error is inevitably contaminated by disturbances in a real system, and should be filtered before the ILC learns it in the current iteration. Let $\tilde{e}_{i-1}(n+1)$ denote the filtered output of $e_{i-1}(n+1)$. According to Eqs. (1) and (2), we have then the CITE-ILC implementation equation

$$u_i(n) = u_{i-1}(n) + L\ \tilde{e}_{i-1}(n+1) + K\ e_i(n) \tag{3}$$

and the SCILC implementation equation

$$u_i(n) = u_{i-1}(n) + L\ \tilde{e}_{i-1}(n+1) + K[e_i(n) - e_{i-1}(n)] \tag{4}$$

respectively. In Eqs. (3) and (4), the current error $e_i(n)$ is not filtered, since such real-time filtering will induce a certain phase delay, which is not desirable in general. Hence, its companion $e_{i-1}(n)$ in Eq. (4) is not filtered either. Fig. 2 is a flowchart of the SCILC algorithm (4), where G_p, H, and z^1 denote the plant, the BSN filter, and the one-step advance element,

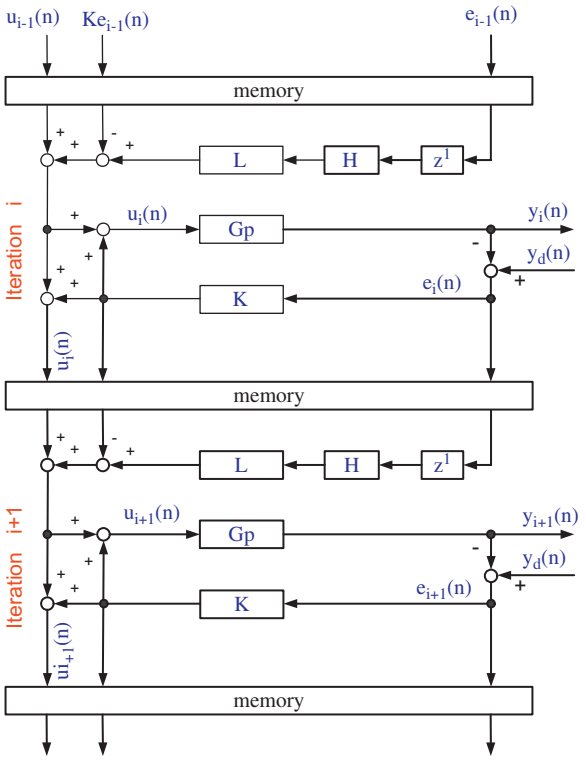

Fig. 2. Flowchart of the SCILC scheme.

respectively. Let the z-transform of a variable $x(n)$ be denoted by $X(z)$. Since the z-transform of $\tilde{e}_{i-1}(n+1)$ is $z\tilde{E}_{i-1}(z)$, we can transform Eq. (4) into the z-domain expression as

$$U_i(z) = U_{i-1}(z) + L\ z\,\tilde{E}_{i-1}(z) + K[E_i(z) - E_{i-1}(z)] \tag{5}$$

Eq. (5) will be further expanded in the end of the next section to generate a complete SCILC equation in the z-domain.

4. B-spline network filtering

Essentially, the BSN in the LFFC is tuned with an iterative method in order that the system output can approach the desired trajectory eventually. A typical time-indexed LFFC system is shown by Fig. 3-a [15], and a second-order BSN is shown by Fig. 3-b [10]. The LFFC system in Fig. 3-a has the control formula

$$u_i(n) = u_i^b(n) + u_i^f(n)$$

where $u_i(n)$, $u_i^b(n)$, and $u_i^f(n)$ are the system control effort, the feedback control (FBC) effort, and the feed-forward control effort at time instant n of iteration i, respectively. The BSN in the LFFC thus maps the time instant n to the feed-forward control $u_i^f(n)$. In Fig. 3-b, each B-spline represents a basis function that has a membership $\mu(n)\in[0,1]$ for the input of time instant n. The triangular B-splines have a support width D corresponding to the bottom width of the triangle, and are equally paced by $D/2$. For convenience, let B denote the half support width, i.e. $B = D/2$. The BSN output is evaluated [10,14] by

$$u_i^f(n) = \sum_{k=0}^{M} \mu_k(n) w_k^i \tag{6}$$

where the splines are numbered from left to right as 0, 1,…,M, respectively, and $w_k{}^i$ denotes the weight of spline k in iteration i. The weights of the splines are tuned by the FBC effort [14] according to

$$\begin{aligned}\Delta w_k^i &= w_k^i - w_k^{i-1} \\ &= \gamma \sum_{m=0}^{N} \mu_k(m) u_{i-1}^b(m)\end{aligned} \tag{7}$$

where γ is the learning rate, and each learning iteration starts from time instant 0 up to N. Combining Eqs. (6) and (7) leads to

$$u_i^f(n) = u_{i-1}^f(n) + \gamma y(n) \tag{8}$$

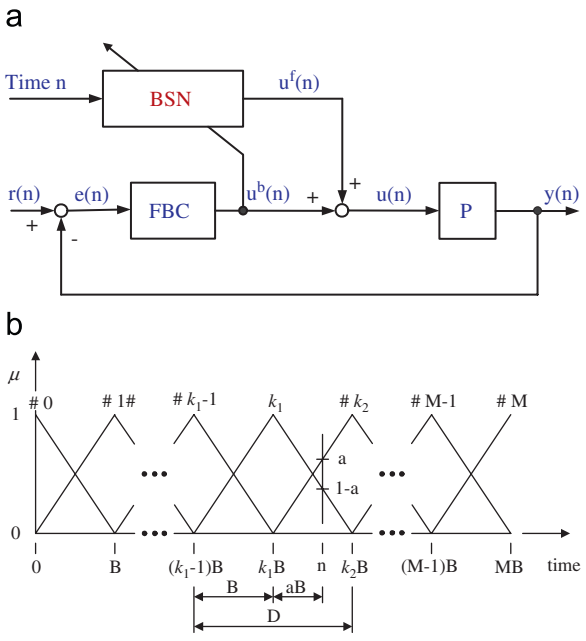

Fig. 3. (a) Time-indexed LFFC system. (b) Second-order B-spline network.

where

$$y(n)=\sum_{k=0}^{M}\mu_k(n)\left[\sum_{m=0}^{N}\mu_k(m)u_{i-1}^{b}(m)\right] \tag{9}$$

From Fig. 3-b, it is observed that at any time instant n there are exactly two splines activated in Eq. (9). If the left and right activated splines are denoted as spline k_1 and k_2, respectively, we have

$$k_1=\lfloor n/B\rfloor,\ k_2=\lceil n/B\rceil \tag{10}$$

where $\lfloor\ \bullet\rfloor$ and $\lceil\ \bullet\rceil$ are flooring and ceiling operators, respectively. We can then reduce Eq. (9) to

$$y(n)=\sum_{k=k_1,k_2}\mu_k(n)\left[\sum_{m=0}^{N}\mu_k(m)u_{i-1}^{b}(m)\right] \tag{11}$$

For convenience, a normalized time fraction a is defined as

$$a=n/B-k_1,\ a\in[0,1] \tag{12}$$

From Fig. 3-b, one gets the memberships of splines k_1 and k_2 at time instant n as

$$\mu_{k_1}(n)=1-a,\ \mu_{k_2}(n)=a \tag{13}$$

respectively. Substituting Eq. (13) into Eq. (11) yields

$$y(n)=(1-a)\sum_{m=0}^{N}\mu_{k_1}(m)u_{i-1}^{b}(m)+a\sum_{m=0}^{N}\mu_{k_2}(m)u_{i-1}^{b}(m) \tag{14}$$

Let the label * denote the convolution operation. According to [10], we can transform Eq. (14) to the filter form as

$$y(n)=[(1-a)z^{-aB}+az^{(1-a)B}]\ \ h_{zp}(n)*u_{i-1}^{b}(n) \tag{15}$$

where $h_{zp}(n)$ is the impulse response of a zero-phase filter which is represented by a symmetrical B-spline basis centered at time instant n, and the z-transform of $h_{zp}(n)$ is

$$H_{zp}(z)=\sum_{l=-B}^{B}\left(1-\frac{|l|}{B}\right)z^{-l} \tag{16}$$

By labeling the digital frequency as f and replacing z in Eq. (16) with $e^{j2\pi f}=e^{j2\pi 0}$, we get the zero-frequency response of $H_{zp}(z)$, and it is equal to B. We can then rewrite Eq. (15) as

$$y(n)=Bh(a,n)*u_{i-1}^{b}(n) \tag{17}$$

where $h(a, n)$ is the impulse response of the BSN filter with unity response at zero frequency, and its z-transform, i.e. the transfer function of the BSN filter in the z-domain, is

$$H(a,z)=[(1-a)z^{-aB}+az^{(1-a)B}]H_{zp}(z)/B \tag{18}$$

Since the BSN filter $H(a, z)$ depends on a, in general it has non-ZP frequency responses. Inserting Eq. (17) into Eq. (8) results in the following LFFC update law based on the second-order BSN:

$$u_i^f(n)=u_{i-1}^{f}(n)+\gamma_d h(a,n)*u_{i-1}^{b}(n)$$

where $\gamma_d=\gamma B$. Finally, by inserting Eq. (16) into Eq. (18) and replacing z with $e^{j2\pi f}$, we obtain the transfer function of the BSN filter in the frequency domain as

$$H(a,f)=\frac{1}{B}[(1-a)e^{-j2\pi f^{aB}}+a\ e^{j2\pi f^{(1-a)B}}]\sum_{l=-B}^{B}\left(1-\frac{|l|}{B}\right)e^{-j2\pi fl} \tag{19}$$

In this research, we use Eq. (18) to filter the learnable error in Eq. (4), i.e.

$$\tilde{e}_{i-1}(n+1)=h(a,n)*e_{i-1}(n+1) \tag{20}$$

By performing the z-transform for Eq. (20), we have

$$\tilde{E}_{i-1}(z)=H(a,z)E_{i-1}(z) \tag{21}$$

Inserting Eq. (21) into Eq. (5) yields the complete SCILC equation in the z-domain as

$$U_i(z) = U_{i-1}(z) + [L\, z\, H(a,z) - K]\, E_{i-1}(z) + K\, E_i(z) \tag{22}$$

Eq. (22) will be used in the next section for convergence analysis of the SCILC.

5. Convergence analysis

The convergence analysis of SCILC (4) can be approached in the frequency domain as follows. In general, an ILC formula can be expressed as

$$u_i(n) = u_{i-1}(n) + G_{ff}\, e_{i-1}(n) + G_{fb}\, e_i(n) \tag{23}$$

where G_{ff} and G_{fb} denote the feed-forward gain and the feedback gain, respectively. We can transform Eq. (23) in the z-domain as

$$U_i(z) = U_{i-1}(z) + G_{ff}\, E_{i-1}(z) + G_{fb}\, E_i(z) \tag{24}$$

Suppose the plant has a linear transfer function $G_p(z)$ such that

$$Y_i(z) = G_p(z)U_i(z),\ Y_{i-1}(z) = G_p(z)U_{i-1}(z) \tag{25}$$

where $Y_i(z)$ and $Y_{i-1}(z)$ denote the system output in iteration i and $i-1$, respectively. Since

$$E_i(z) = Y_d(z) - Y_i(z),\ E_{i-1}(z) = Y_d(z) - Y_{i-1}(z)$$

where $Y_d(z)$ is the desired output, and $E_i(z)$ and $E_{i-1}(z)$ are the errors in iteration i and $i-1$, respectively, we can rewrite Eq. (25) as

$$Y_d(z) - E_i(z) = G_p(z)U_i(z),\ Y_d(z) - E_{i-1}(z) = G_p(z)U_{i-1}(z) \tag{26}$$

By multiplying Eq. (24) by $G_p(z)$ and referring to Eq. (26), we obtain

$$\frac{E_i(z)}{E_{i-1}(z)} = \frac{1 - G_{ff}G_p(z)}{1 + G_{fb}G_p(z)} \tag{27}$$

Replacing z in Eq. (27) with $e^{j2\pi f}$ yields the frequency domain expression

$$\rho(f) = \left|\frac{E_i(f)}{E_{i-1}(f)}\right| = \left|\frac{1 - G_{ff}G_p(f)}{1 + G_{fb}G_p(f)}\right| \tag{28}$$

where $\rho(f)$ is defined as the convergence rate. The condition of error convergence is

$$\rho(f) = \left|\frac{E_i(f)}{E_{i-1}(f)}\right| = \left|\frac{1 - G_{ff}G_p(f)}{1 + G_{fb}G_p(f)}\right| < 1,\ \forall f \tag{29}$$

Now, by comparing Eq. (22) with Eq. (24) and replacing z with $e^{j2\pi f}$, we have

$$G_{ff} = L\, e^{j2\pi f}\, H(a,f) - K,\ G_{fb} = K \tag{30}$$

Substituting Eq. (30) into Eq. (29) leads to

$$\rho(a,f) < 1,\ \ \forall f,\ \ \forall a \tag{31}$$

$$\rho(a,f) = \left|\frac{1 - [L\ e^{j2\pi f}\, H(a,f) - K]\ G_p(f)}{1 + K\, G_p(f)}\right| \tag{32}$$

where the convergence rate is re-labeled as $\rho(a,f)$, since it is now a function of both the normalized time fraction a and the frequency f.

In many situations, however, Eq. (31) may not hold for all frequencies. Instead, the inequality may be valid only for frequencies below a certain threshold. Hence, we define a convergence bandwidth f_b as the threshold frequency that satisfies

$$\rho(a,f_b) = 1,\ \forall a \tag{33}$$

or, by referring to Eq. (32),

$$\rho(a,f_b) = \left|\frac{1 - [L\, e^{j2\pi f_b}\, H(a,f_b) - K]\, G_p(f_b)}{1 + K\, G_p(f_b)}\right| = 1,\ \forall a \tag{34}$$

The convergence bandwidth varies with a, and should be labeled as $f_b(a)$ indeed. If Eq. (34) has multiple real roots, then $f_b(a)$ is the minimal positive root; if Eq. (34) has no real root, $f_b(a)$ is assigned as the Nyquist frequency. For error convergence, one should select a proper support width for the BSN filter such that its cutoff frequency $f_c(a)$ is less than the

convergence bandwidth $f_b(a)$ for any a, i.e.

$$f_c(a) < f_b(a), \ \forall a \tag{35}$$

If the system has the property that the signal components of the tracking errors all reside in the frequencies below the cutoff $f_c(a)$ for any a, then the selection of Eq. (35) ensures that all signal components of the errors will be learned and converge to zero eventually. Therefore, by using Eq. (35), tracking errors will finally diminish to the system noise level.

6. Time-frequency analysis

On the condition of Eq. (35), we can select a proper cutoff frequency such that the signal components of the tracking errors are retrieved whereas the noise components are filtered out. Distinguishing the signal from the noise can be accomplished by analysis of the errors. To analyze non-stationary data such as the tracking errors, the short-time Fourier transform (STFT), wavelet transform (WT), and Wigner–Ville distribution (WVD) are typical methods. In general, the WVD can characterize the time-dependent spectra of the data better than the STFT and WT [17], but it has the drawback of cross-term interference. Nevertheless, the interference problem can be treated by WVD variations, such as the notable smoothed pseudo WVD [17]. Referring to the ILC study using the time-frequency analysis in [9], we adopted the WVD to analyze the tracking errors. Some properties of the WVD are stated in the following.

The WVD of a signal $x(t)$ is defined as

$$W_x(t,F) = \int_{-\infty}^{\infty} x\left(t+\frac{\tau}{2}\right)x^*\left(t-\frac{\tau}{2}\right)e^{-j2\pi F\tau}\,d\tau \tag{36}$$

where t, F, and * denote the analog time, the analog frequency, and the conjugate of a variable, respectively. By using $W_x(t, F)$, the instantaneous frequency (IF) of $x(t)$ can be recovered as

$$F_x(t) = \frac{\int_{-\infty}^{\infty} W_x(t,F)F\,dF}{\int_{-\infty}^{\infty} W_x(t,F)\,dF} \tag{37}$$

Due to the frequency marginal property of the WVD, the energy density spectrum (EDS) of $x(t)$ can be obtained by

$$|X(F)|^2 = \int_{-\infty}^{\infty} W_x(t,F)\,dt \tag{38}$$

Further, the smoothed pseudo WVD of $x(t)$, $W_x'(t, F)$, is given by

$$W'_x(t,F) = \int_{\infty}^{\infty} w(\tau)\left[\int_{-\infty}^{\infty} v(t-t')x\left(t'+\tfrac{\tau}{2}\right)x^*\left(t'-\tfrac{\tau}{2}\right)dt'\right] e^{-j2\pi F\tau}\,d\tau \tag{39}$$

where $w(\cdot)$ and $v(\cdot)$ are smoothing functions such as the Hamming windows used to eliminate the cross-term interference problem of the WVD analysis. Eqs. (36)–(39) will be used in the next section for the time-frequency analysis of tracking errors in order to determine the cutoff frequency and the support width of the BSN filter.

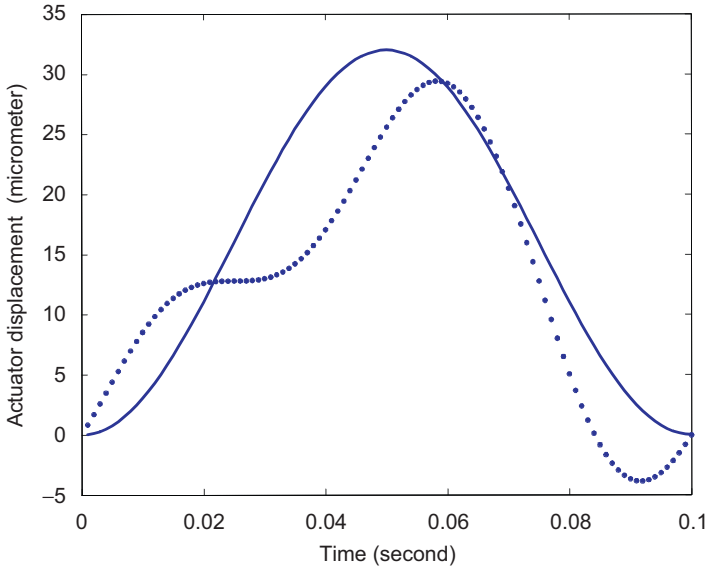

Fig. 4. Desired tracking profiles: a single-sinusoid trajectory (the solid curve), and a multiple-sinusoid trajectory (the dotted curve).

7. Experiments

Two desired trajectories as shown in Fig. 4 were the tracking targets for implementing the SCILC scheme. Both trajectories had a stroke length of 32 μm and a cycle time of 0.1 s. Each trajectory was tracked five cycles per iteration in the experiment, and thus had a frequency of 10 Hz. The solid curve in Fig. 4 represents one cycle of the first trajectory which was a single sinusoid of 10 Hz frequency. The dotted curve in Fig. 4 depicts one cycle of the second trajectory which consisted of two sinusoids: a fundamental sinusoid of 10 Hz frequency and its second harmonic of 20 Hz frequency. For convenience, the first and second trajectories are named as the single-sinusoid trajectory and the multiple-sinusoid trajectory, respectively, in what follows.

Based on a sampling rate of about 1000 samples per second according to the experimental setup, there were about 100 samples in each trajectory. Besides the initial iteration required for starting the learning process, there were 50 learning iterations in each experiment. The design of the iterative learning controller is described as follows.

7.1. Iterative learning controller design

In this design, the digital frequency f (cycles/sample) is adopted as the default frequency. The analog frequency F (Hz) is equal to the digital frequency f times the sampling rate S (samples/s). Since S is about 1000 samples per second in the experiments, F is thus equal to $1000f$ approximately.

7.1.1. Learning gain and proportional gain

To simplify our experiments, we equalized the proportional gain K and the learning gain L, i.e. $K = L$. From the ratio of the sensor amplifier output -1 V maximum to the power amplifier input +5 V maximum, we chose $-1/5$ as the nominal gain g of the system. We then assigned g^{-1}, i.e. -5, to both the learning gain and the proportional gain as a reasonable value for use in the experiments.

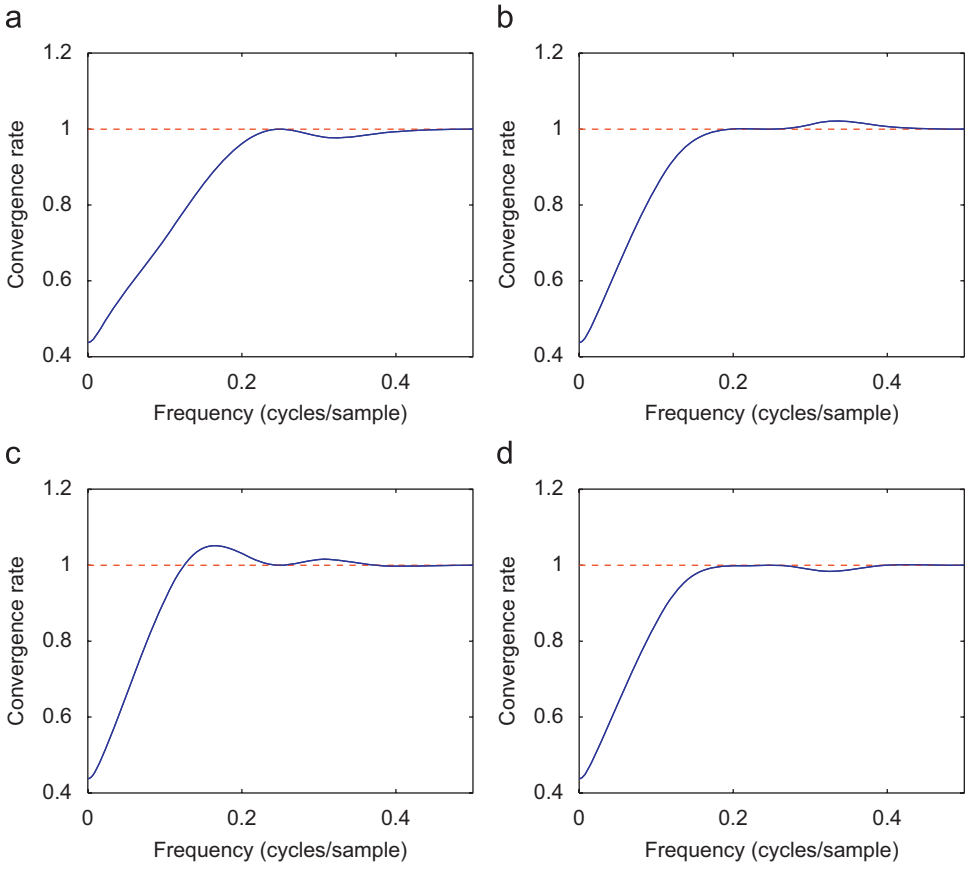

Fig. 5. Normal plots of the convergence rate $\rho(a, f)$ for (a) $a = 0$, (b) $a = 0.25$, (c) $a = 0.5$, and (d) $a = 0.75$, respectively.

7.1.2. Convergence bandwidth of the SCILC system

According to Eq. (32), the convergence rate $\rho(a,f)$ depends on L, K, $H(a,f)$, and $G_p(f)$. The gains L and K have been assigned -5 as above. The frequency response of the filter $H(a, f)$ is given by Eq. (19). Here, the transfer function $G_p(f)$ can be estimated as

$$\hat{G}_p(f) = \frac{-0.0823e^{-j2\pi f} - 0.0068e^{-j4\pi f} + 0.0564e^{-j6\pi f}}{1 - 0.873e^{-j2\pi f}}$$

by using a system identification technique [7]. Hence, we can obtain the convergence rate $\rho(a, f)$ by using Eq. (32). Fig. 5 gives four normal plots of $\rho(a, f)$ for a equal to 0, 0.25, 0.5 and 0.75, respectively. In each sub-figure, the solid curve represents $\rho(a, f)$ while the dashed line means the unity value, and the first intersection of the two traces determines the convergence bandwidth for the corresponding a. Fig. 6 shows the zoom-in plots of $\rho(a, f)$, from which one can observe clearly the intersections of the $\rho(a, f)$ curves with the unity lines for different a values. By using Eq. (34), we obtain the convergence bandwidth $f_b(a)$ of the SCILC system at 0.249, 0.196, 0.124, and 0.249, corresponding to the values of a equal to 0, 0.25, 0.5, and 0.75, respectively. Fig. 7 shows the plots of the convergence bandwidth $f_b(a)$ versus the normalized time fraction a for the half spline support B equal to 3, 4, 5, and 6, respectively. It can be observed from the figure that the minimum $f_b(a)$ occurs when a is equal to 0.5, and it is 0.17, 0.12, 0.10, and 0.08, corresponding to the values of B equal to 3, 4, 5, and 6, respectively. The plots of the convergence bandwidth in Fig. 7 will be used in sub-section 7.1.4 for determining the cutoff frequency and support width of the BSN.

7.1.3. Wigner–Ville distribution of tracking errors

For a controlled system with recorded tracking errors of past iterations, one can analyze the frequency contents of the errors using Eqs. (36)–(39) [8]. Fig. 8 shows the typical frequency contents of the tracking errors occurring in this experimental system. The subplots in the figure show (a) a typical tracking error record, (b) an estimation of its instantaneous frequency, (c) an estimation of its energy density spectrum, and (d) its smoothed pseudo WVD, respectively. From Fig. 8, one can observe that the tracking errors of the experimental system resided in two separate frequency bands: one low-frequency band which was below a frequency of about 0.08, and the other high-frequency band which was above a frequency of roughly 0.15. Hence, it can be reasonably assumed that the low-band spectra were correlated to input signals

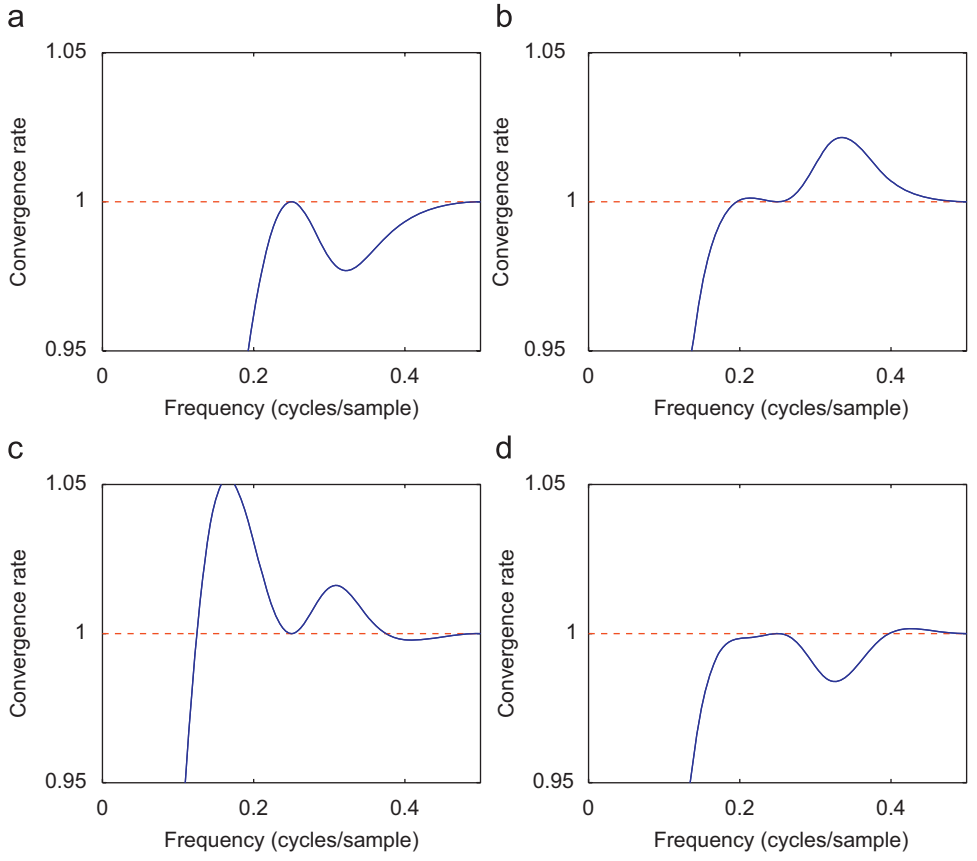

Fig. 6. Zoom-in plots of the convergence rate $\rho(a, f)$ for (a) $a = 0$, (b) $a = 0.25$, (c) $a = 0.5$, and (d) $a = 0.75$, respectively.

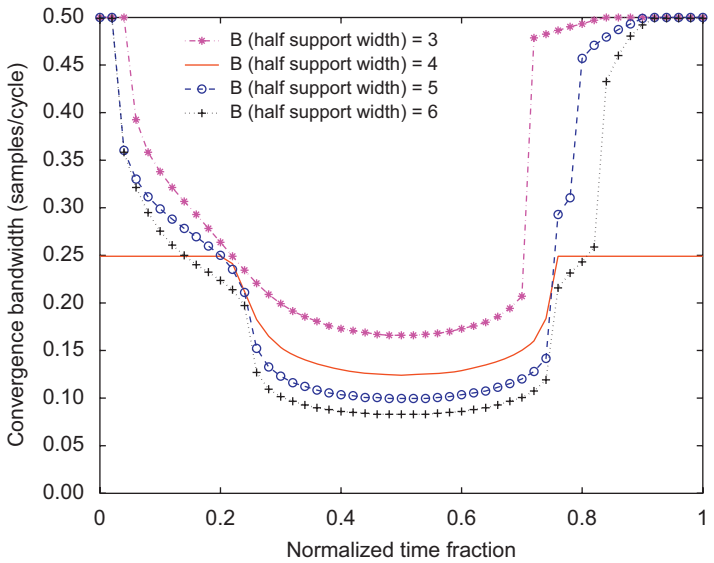

Fig. 7. Convergence bandwidth $f_b(a)$ versus normalized time fraction a for the BSN-based ILC system using the half support width B of 3, 4, 5, and 6, respectively.

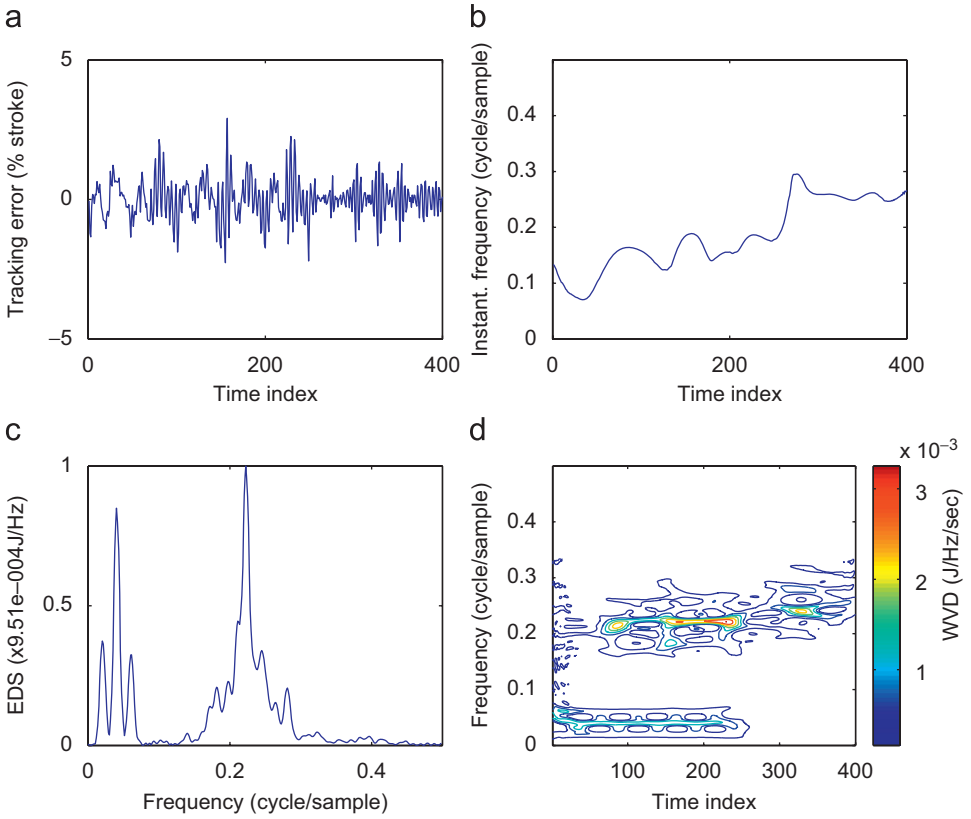

Fig. 8. WVD analysis of tracking errors in a learning iteration: (a) a record of error measurement, (b) an estimation of the instantaneous frequency, (c) an estimation of the energy density spectrum, and (d) the smoothed pseudo WVD.

to the system, whereas the high-band spectra were caused by system disturbances. Therefore, one can select a proper support width for the BSN filter such that it has a cutoff frequency $f_c(a)$ being above 0.08, but below the convergence bandwidth $f_b(a)$ according to Eq. (35), for any a.

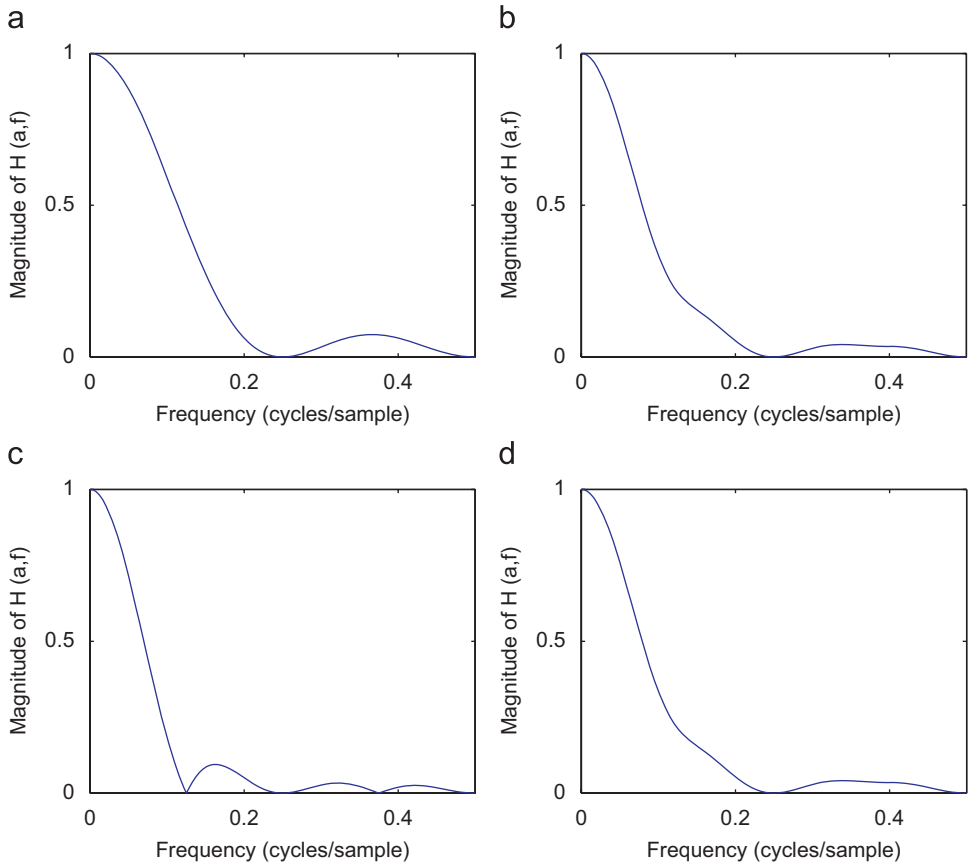

Fig. 9. Magnitude responses of the BSN filter $H(a, f)$ for (a) $a = 0$, (b) $a = 0.25$, (c) $a = 0.5$, and (d) $a = 0.75$, respectively. The half support width of the BSN is 4.

7.1.4. Support width of the BSN

Given a support width D or a half support width B of the BSN, one can use Eq. (19) to acquire the frequency response of the BSN filter, and thus get the cutoff frequency $f_c(a)$ of that filter for any normalized time fraction a. For example, given B equal to 4, one can thereby obtain the magnitude responses of the filter as shown in Fig. 9, and the phase responses of the filter as shown in Fig. 10, for a equal to 0, 0.25, 0.5, and 0.75, respectively. In Fig. 9, by adopting -8 dB attenuation as the cutoff threshold [10], the cutoff frequencies of the filter are 0.11, 0.078, 0.069, and 0.078 for a equal to 0, 0.25, 0.5, and 0.75, respectively. Further, since the phase delay of a filter is determined by the negative derivative of its phase response, one can observe from Fig. 10-a and -c that the BSN filter has zero-phase delay when a is equal to 0 and 0.5, respectively. Moreover, according to Fig. 10-b, the filter has positive phase delay (i.e. phase lag) for some frequencies when a is equal to 0.25. On the other hand, from Fig. 10-d, the filter has negative phase delay (i.e. phase lead) for some frequencies when a is equal to 0.75. This means that the BSN filter is not ZP in general, except for some distinct a values.

Fig. 11 gives the plots of $f_c(a)$ versus a for B equal to 3, 4, 5, and 6, respectively. It is observed that $f_c(a)$ has a minimum $f_{c,min}$ on a equal to 0.5 and a maximum $f_{c,max}$ on a equal to 0 or 1. Indeed, $f_{c,min}$ and $f_{c,max}$ are 0.10 and 0.18 for B equal to 3; 0.08 and 0.13 for B equal to 4; 0.06 and 0.10 for B equal to 5; 0.05 and 0.09 for B equal to 6, respectively. By comparing Fig. 11 with Fig. 7, it is obvious that the convergence condition (35) is satisfied for all B values. Hence the selection of B is subject to the criterion that the cutoff frequency $f_b(a)$ should be above 0.08 and below 0.15, in accordance with the WVD analysis of tracking errors in the previous sub-section, for any a. Therefore, judging from Fig. 11, it is proper to select 4 for the half support width B, or 8 for the support width D, of the BSN.

In summary, the BSN filter (19) was determined by selecting the half support width of 4. The filter then served the filtering of learnable tracking errors for the SCILC (4) and CITE-ILC (3) schemes. Both schemes used the same gain of -5, and were applied to track the desired trajectories, respectively, for comparison. For presentation compactness, we will use the terms SCILC and CITE-ILC in the next sub-section to represent the SCILC and CITE-ILC schemes based on the designed BSN filter, respectively. The experimental results accompanied by their tracking performances are presented in the following.

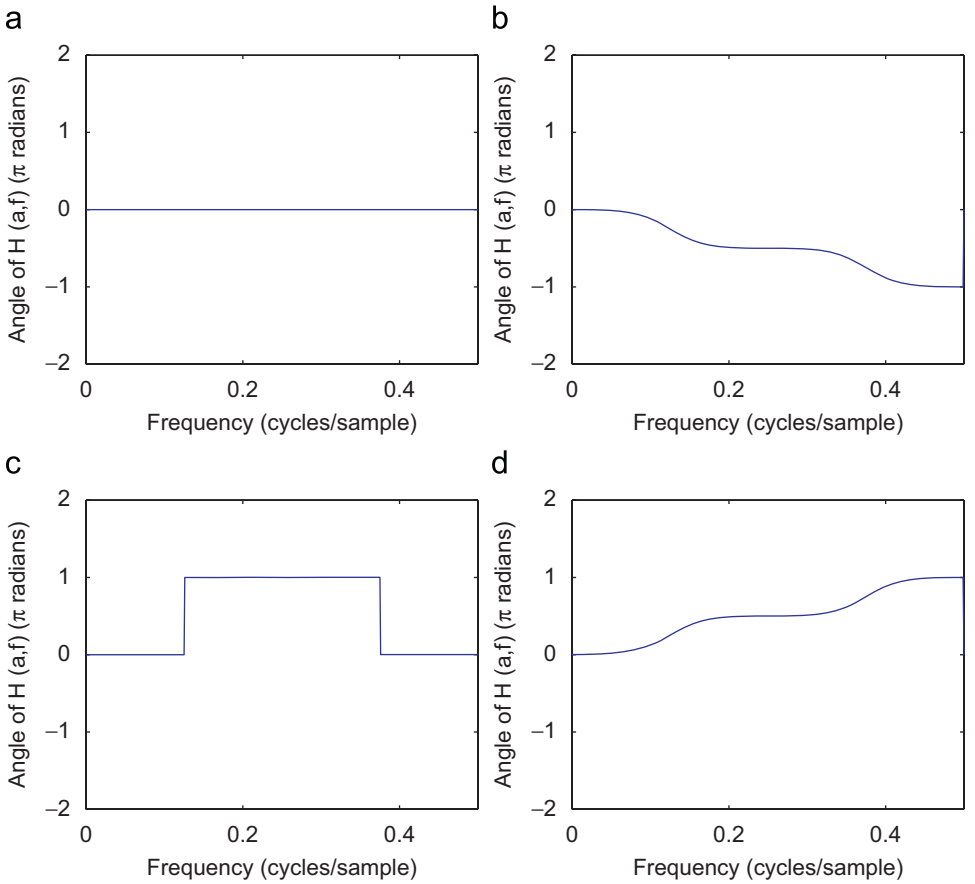

Fig. 10. Phase responses of the BSN filter $H(a, f)$ for (a) $a = 0$, (b) $a = 0.25$, (c) $a = 0.5$, and (d) $a = 0.75$, respectively. The half support width of the BSN is 4.

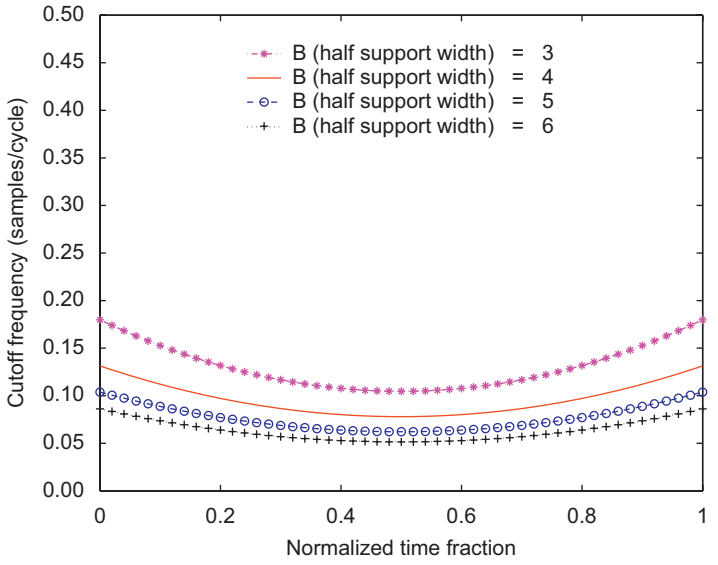

Fig. 11. Cutoff frequency of the BSN filter $f_c(a)$ versus normalized time fraction a for the half support width B equal to 3, 4, 5, and 6, respectively.

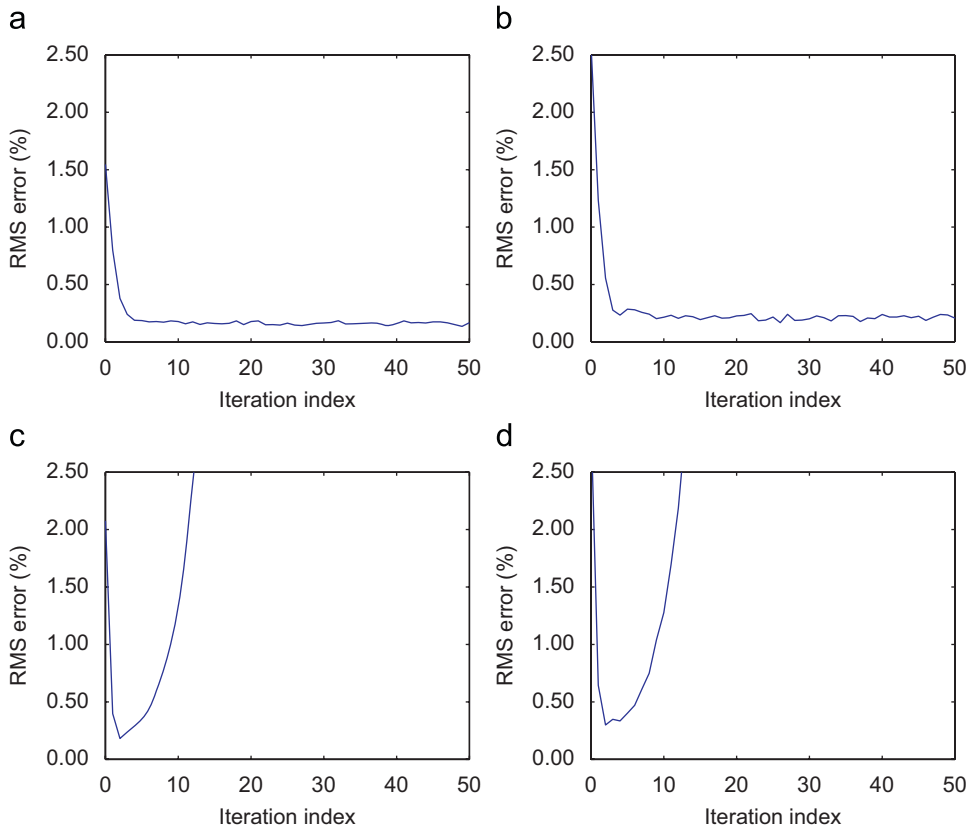

Fig. 12. RMS errors vs. iteration index: (a) using the SCILC to track the single-sinusoid trajectory, (b) using the SCILC to track the multiple-sinusoid trajectory, (c) using the CITE-ILC to track the single-sinusoid trajectory, and (d) using the CITE-ILC to track the multiple-sinusoid trajectory.

7.2. Experimental results

Fig. 12 shows the experimental results of (a) using the SCILC to track the single-sinusoid trajectory of 10 Hz and 32 μm, (b) using the SCILC to track the multiple-sinusoid trajectory of 10 Hz and 32 μm, (c) using the CITE-ILC to track the single-sinusoid trajectory of 10 Hz and 32 μm, and (d) using the CITE-ILC to track the multi-sinusoid trajectory of 10 Hz and 32 μm, respectively. In Fig. 12-a, the tracking errors converged within several iterations, and the steady-state RMS errors were about 0.16% (0.052 μm) of the stroke length. Likewise in Fig. 12-b, the errors converged quickly and had steady-state values at about 0.21% (0.068 μm) of the stroke length. In Fig. 12-c and -d, however, the errors diverged after a couple of iterations. It is evident that the SCILC outperforms the CITE-ILC substantially in tracking performance.

Considering the strain gauge accuracy and the ADC and DAC accuracies for the experimental setup, the system noise level was estimated to be about 0.15% of the stroke length. Hence, on trajectory tracking of the piezoelectric actuator system, the control scheme presented in this paper proves to be effective in achieving fast error convergence for the system, and the long-term stability is retained with its tracking errors close to the background noise level. Furthermore, according to our experiments, the SCILC could yield similar results for the controlled piezoelectric actuator system even though the learning and proportional gains were varied by the amount of ± 2. This implies that the proposed SCILC controller is quite robust in terms of gain variations.

In order to evaluate the overall performance of the proposed SCILC, the controller for the piezoelectric actuator using the same control parameters was further employed to track the two desired trajectories for different frequencies and stroke lengths. Fig. 13 gives the results of using the SCILC to track the single-sinusoid trajectories with their frequencies and strokes as (a) 5 Hz and 16 μm, (b) 5 Hz and 40 μm, (c) 20 Hz and 16 μm, and (d) 20 Hz and 40 μm, respectively. The obtained steady-state RMS errors are (a) 0.25% (0.040 μm) for 5 Hz and 16 μm, (b) 0.11% (0.046 μm) for 5 Hz and 40 μm, (c) 0.29% (0.047 μm) for 20 Hz and 16 μm, and (d) 0.22% (0.089 μm) for 20 Hz and 40 μm, respectively. Fig. 14 shows the results of using the SCILC to track the multiple-sinusoid trajectories for their frequencies and strokes of (a) 5 Hz and 16 μm, (b) 5 Hz and 40 μm, (c) 10 Hz and 16 μm, and (d) 10 Hz and 40 μm, respectively. The steady-state RMS errors acquired are (a) 0.32% (0.051 μm) for 5 Hz and 16 μm, (b) 0.15% (0.058 μm) for 5 Hz and 40 μm, (c) 0.34% (0.054 μm) for 10 Hz and 16 μm, and (d) 0.19% (0.077 μm) for 10 Hz and 40 μm, respectively.

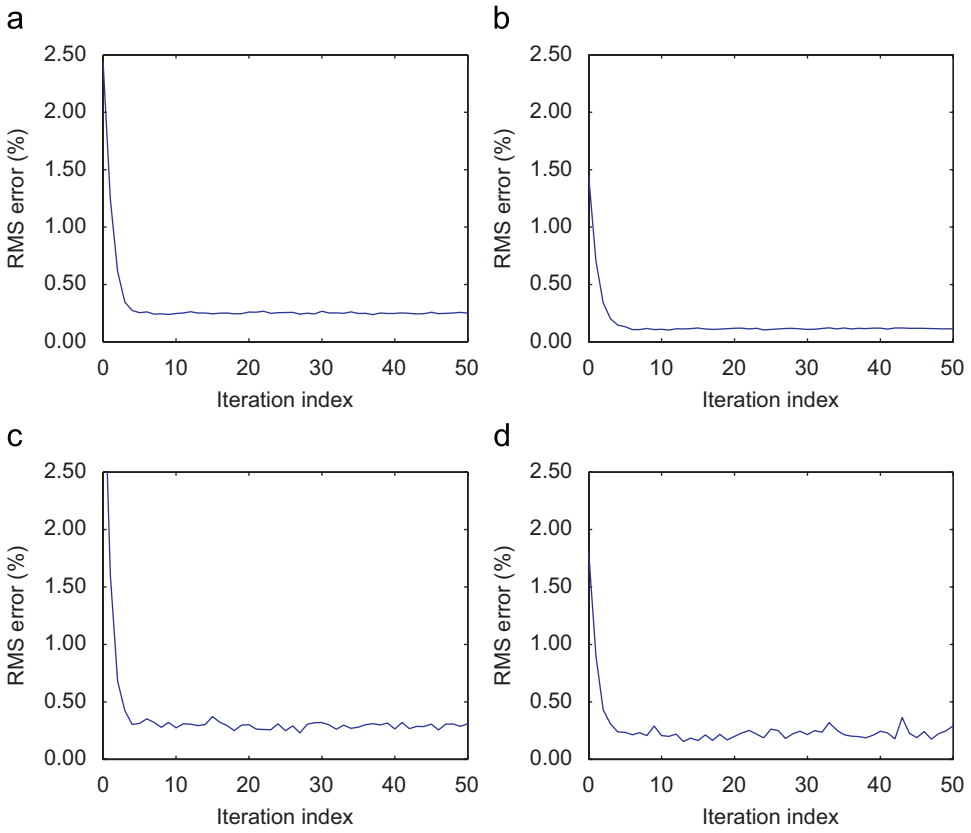

Fig. 13. RMS errors vs. iteration index using the SCILC to track the single-sinusoid trajectory for different frequencies and strokes: (a) 5 Hz and 16 μm; (b) 5 Hz and 40 μm; (c) 20 Hz and 16 μm; and (d) 20 Hz and 40 μm.

From the observations on Figs. 12–14, the SCILC could obtain similar performances in tracking the desired trajectories for certain ranges of different frequencies and strokes of the piezoelectric actuator. It indicates that the SCILC method is consistent under the different frequencies of trajectory tracking. The consistency of the method under different frequencies can be guaranteed by the fact that the control program of the method is designed using the digital period of the trajectory, and can therefore apply to different frequencies of trajectory tracking. The limits of the SCILC method are, however, subject to the limitation of the experimental platform. For example, when the frequency of a tracked single-sinusoid trajectory was raised up to higher than 20 Hz, tracking errors deteriorated since the digital period was then smaller than 50 samples per trajectory cycle. It is obvious that a smaller digital period will result in bigger tracking errors by nature of a typical digital control system.

8. Conclusion

This research applied the BSN filtering to a state-compensated ILC scheme for trajectory tracking of a piezoelectric actuator. The ILC filtering in the literature is usually based on a finite impulse response (FIR) or infinite impulse response (IIR) design which aims at sharp cutoff and zero-phase filtering. On the contrary, the BSN filtering is apart from such FIR or IIR design approach in terms of the cutoff sharpness and the zero-phase response. Nevertheless, as revealed by the experimental results for the piezoelectric actuator, the SCILC scheme using BSN filtering could still achieve fast error convergence and small steady-state errors in tracking the desired trajectories for wide ranges of frequencies and amplitudes. The BSN filtering, therefore, relaxes the filter design criterion of zero-phase delay traditionally adopted in the ILC literature. This paper also presents the detailed procedures of the controller design for the SCILC scheme based on BSN filtering. The convergence bandwidth analysis of the system using the BSN-based SCILC scheme is discussed theoretically and then applied to the controller design. The transfer function of the BSN filter is also derived. The formula of the transfer function is then employed to determine the cutoff frequency and the support width of the filter, with the aid of the time-frequency analysis of the tracking errors using the Wigner–Ville distribution method. The clearly described procedures of the controller design are appealing for other applications using the SCILC scheme based on BSN filtering.

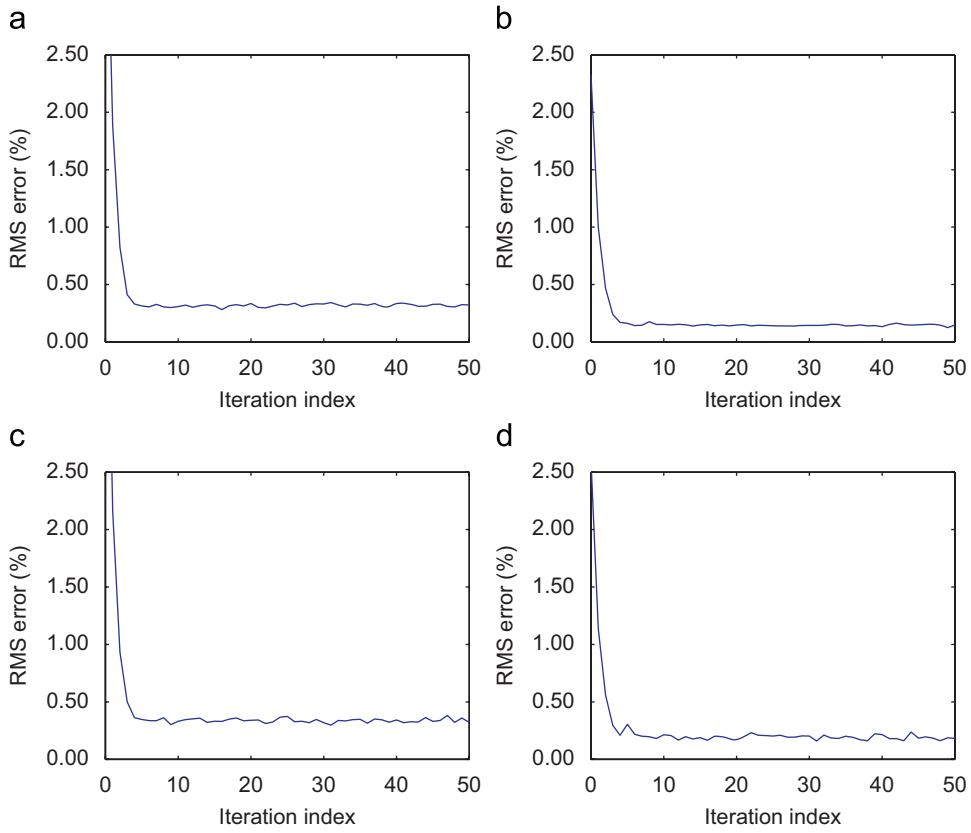

Fig. 14. RMS errors vs. iteration index using the SCILC to track the multiple-sinusoid trajectory for different frequencies and strokes: (a) 5 Hz and 16 μm; (b) 5 Hz and 40 μm; (c) 10 Hz and 16 μm; and (d) 10 Hz and 40 μm.

Acknowledgments

This work was supported by the National Science Council, ROC., under grant NSC93-2218-E-211-002 and NSC93-2212-E-211-007.

References

[1] P. Ge, M. Jouaneh, Tracking control of a piezoceramic actuator, IEEE Transactions on Control Systems Technology 4 (3) (1996) 209–216.
[2] C.L. Hwang, C. Jan, A reinforcement discrete neuro-adaptive control for unknown piezoelectric actuator systems with dominant hysteresis, IEEE Transactions on Neural Networks 14 (1) (2003) 66–78.
[3] Z. Bien, J.X. Xu, Iterative Learning Control: Analysis, Design Integration and Application, Kluwer Academic Publishers, Boston, MA, 1998.
[4] K. Leang, S. Devasia, Iterative feedforward compensation of hysteresis in piezo positioners, in: Proceedings of the 2003 IEEE Conference on Decision and Control, Maui, Hawaii, 2003, pp. 2626–2631.
[5] Y.C. Huang, C.H. Cheng, Robust tracking control of a novel piezodriven monolithic flexure-hinge stage, in: Proceedings of the 2004 IEEE Conference on Control Applications, Taipei, Taiwan, 2004, pp. 977–982.
[6] J.X. Xu, J. Xu, T. Lee, Iterative learning control for a linear piezoelectric motor with a nonlinear unknown input deadzone, in: Proceedings of the 2004 IEEE Conference on Control Applications, Taipei, Taiwan, 2004, pp. 1001–1006.
[7] F.S. Lee, C.J. Chien, J.C. Wang, J.J. Liu, Application of a model-based iterative learning technique to tracking control of a piezoelectric system, Asian Journal of Control 7 (1) (2005) 29–37.
[8] C.J. Chien, F.S. Lee, J.C. Wang, Trajectory tracking of a piezoelectric system using state compensated iterative learning control, in: Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference ECC 2005, Seville, Spain, December 12–15, 2005, pp. 5826–5831.
[9] Y. Chen, K. Moore, Frequency domain adaptive learning feedforward control, in: Proceedings of the 2001 IEEE International Symposium on Computational Intelligence in Robotics and Automation, Alberta, Canada, 2001, pp. 396–401.
[10] Y. Chen, K. Moore, V. Bahl, Learning feedforward control using a dilated B-spline network: frequency domain analysis and design, IEEE Transactions on Neural Networks 15 (2) (2004) 355–366.
[11] J. Starrenburg, W. van Luenen, W. Oelen, J. Van Amerongen, Learning feedforward controller for a mobile robot vehicle, Control Engineering Practice 14 (9) (1996) 1221–1230.
[12] W. Velthuis, T. de Vries, J. Van Amerongen, Learning feedforward control of a flexible beam, in: Proceedings of the 1996 IEEE International Symposium on Intelligent Control, Dearborn, MI, 1996, pp. 104–108.
[13] G. Otten, T. De Vries, J. Van Amerongen, A. Rankers, E. Gaal, Linear motor motion control using a learning feedforward controller, IEEE/ASME Transactions on Mechatronics 2 (3) (1997) 179–187.

[14] W. Velthuis, T. De Vries, E. Gaal, Experimental verification of the stability analysis of learning feed-forward control, in: Proceedings of the 37th IEEE Conference on Decision Control, Tampa, FL, 1998, pp. 1225–1229.
[15] T. De Vries, W. Velthuis, J. Van Amerongen, Learning feedforward control: a survey and historical note, in: Proceedings of the First IFAC Conference on Mechatronic Systems, Darmstadt, Germany, 2000, pp. 949–954.
[16] C.J. Chien, A discrete iterative learning control for a class of nonlinear time-varying systems, IEEE Transactions on Automatic Control 43 (5) (1998) 748–752.
[17] S. Qian, D. Chen, Joint Time-Frequency Analysis: Methods and Applications, Prentice Hall, Upper Saddle River, NJ, 1996.