Wavelet Transform-Based Frequency Tuning ILC

Bin Zhang, *Student Member, IEEE*, Danwei Wang, *Member, IEEE*, and Yongqiang Ye

***Abstract*—In this paper, a discrete wavelet transform-based cutoff frequency tuning method is proposed and experimental investigation is reported. In the method, discrete wavelet packet algorithm, as a time-frequency analysis tool, is employed to decompose the tracking error into different frequency regions so that the maximal error component can be identified at any time step. At each time step, the passband of the filter is from zero to the upper limit of frequency region where the maximal error component resides. Hence, the filter is a function of time as well as index of cycle. The experimental results show that this method can suppress higher frequency error components at proper time steps. While at the time steps where the major tracking error falls into lower frequency range, the cutoff frequency of the filter is set lower to reduce the influence of noises and uncertainties. This way, learning transient and long-term stability can be improved.**

***Index Terms*—Cutoff frequency tuning, discrete wavelet packet algorithm, distribution index, iterative learning control (ILC).**

I. INTRODUCTION

ITERATIVE learning control (ILC) is very effective to improve the performance of systems that carry out same tasks repeatedly. Its objective is to get zero tracking error as operation goes to infinity, and during this process, keep good learning transient and convergence rate. In manufacturing applications, chemical industry, aerospace industry etc., there are many such systems where ILC is a very promising application.

In the mid-1980s, Arimoto *et al.* rigorously formulated the problem of ILC [1]. Other independent precursors include Casalino *et al.* [2], Craig [3], and Middleton *et al.* [4]. The early work of ILC are mainly in time domain because the learning process is intended for a fixed finite time interval and its analysis results can be easily extended to time-varying and nonlinear systems [5]. The limitation is that time domain analysis does not give useful frequency domain insights of learning. In addition, the time-domain analysis result does not address the issue of good transients and long-term stability.

To improve learning performance, the first thing to consider in time domain is to adjust learning gain. Chang *et al.* [6] pointed out that the tuning of learning gain on iteration axis requires much system knowledge to guarantee good learning transient and this makes implementation difficult. Lee *et al.* [7] proposed a learning gain changing scheme on time axis to get monotonic learning transient in the sense of ∞-norm. Although a learning gain changing scheme makes sense in analysis, Wirkander *et al.* [8] pointed out that learning gain is not a critical factor to learnable bandwidth. Then, for a learning system expecting a well-behaved learning transient and good tracking error level, the result of this scheme is often not obvious and sometimes it even cannot work.

Recently, more and more research efforts turn to frequency response methods [9]–[13]. Tang *et al.* [14] designed a learning controller to individually control each harmonic components of actual output based on Fourier analysis. This equals to handling error components separately, which is reported outperforming conventional ILCs. Zhang *et al.* proposed a cutoff-frequency phase-in method [15]. Adaptive schemes of cutoff frequency are also proposed in frequency domain [11], [16]–[18] to improve performance. In [16], an iteration varying filter method is presented but the performance of this scheme heavily depends on system model. In [11], [17], [18], continuous Wigner transform is used to analyze the signal. Chen *et al.* [11] is a pioneering work in introducing time-frequency domain analysis into ILC. They proposed an adaptive scheme of learning feedforward control based on a B-spline network. Zheng [17] and Ratariu *et al.* [18] used an adaptive Q-filter, which is a moving average filter. In [19], we propose using wavelet transform for time-frequency analysis and design of ILC. Xu [20] used wavelet network in ILC but his work was in time domain to deal with uncertainties.

In this paper, a cutoff frequency tuning method is proposed based on time-frequency analysis and some experimental results are presented to verify the method. In our method, at each time step error components on different frequencies can be identified by using discrete wavelet packet decomposition. Then, based on frequency content of error, cutoff frequency of the filter at each time step can be set accordingly to cover the main error components. This method can let higher frequency error components enter learning at proper time steps and suppress them. At the same time, learning transient and long-term stability can be improved because at other time steps, the cutoff frequency of the filter is lower so that the effect of high-frequency noise and uncertainties can be minimized.

The paper is organized as follows. In Section II, the wavelet packet algorithm is briefly introduced. Then, the cutoff frequency tuning scheme is discussed in detail in Section III, which is followed by some experimental results on a SCARA robot in Section IV. Finally, concluding remarks are given in Section V.

II. WAVELET PACKET ALGORITHM FOR ERROR ANALYSIS

Most signals are in time-domain. To get the frequency domain information of signals, discrete Fourier transform (DFT) is often employed. One disadvantage of Fourier transform is that it will lose time information in frequency domain. To keep both time

Manuscript received March 24, 2004; revised August 24, 2004. This paper was recommended by Associate Editor J. Wang.

B. Zhang and D. Wang are with the School of Electrical and Electronics Engineering, Nanyang Technological University, Nanyang, Singapore 639798 (e-mail: binzhang@pmail.ntu.edu.sg; edwwang@ntu.edu.sg).

Y. Ye was with the School of Electrical and Electronics Engineering, Nanyang Technological University, Nanyang, Singapore 639798. He is now with the School of Information, Zhejiang Institute of Finance and Economics, Hangzhou 310012, China (e-mail: yongqiang_leaf@hotmail.com).

Digital Object Identifier 10.1109/TSMCB.2004.841411

and frequency information, wavelet transform is developed [21]. In this transform, a fully scalable modulated window, referred to as a wavelet, is a waveform of effectively limited duration that has an average value of zero. This window is shifted along the time axis and the signal spectrum is calculated at every time step. After that, a window of slightly different width is used to repeat this process. In the end of the processing, a collection of time-frequency representation of the signal with different resolution will be obtained. The result is referred to as multiresolution analysis (MRA). It can give us time information and frequency information simultaneously with desired resolution.

A. Wavelet Packet Algorithm

For the space $L^2(R)$ of all square integrable functions, multiresolution analysis is defined as a sequence of closed subspaces V_j of $L^2(R)$ for $j \in Z = 0, 1, 2, \ldots$. V_j is spanned by the family

$$\phi_{j,k}(x) = 2^{j/2}\phi(2^j x - k); \quad k \in Z \tag{1}$$

with ϕ being a scaling function. The space $L^2(R)$ is a closure of the union of all V_j. The sequence of subspaces V_j is nested, i.e., $V_j \subset V_{j+1}$. Moreover, it has features $f(x) \in V_j \iff f(2x) \in V_{j+1}$ and $f(x) \in V_j \iff f(x+k) \in V_j; k \in Z$. If the space V_j is spanned by functions $\phi_{j,k}(x)$, then space V_{j+1} is spanned by $\phi_{j+1,k} = \sqrt{2}\phi_{j,k}(2x)$.

Because $V_j \subset V_{j+1}$, any function in V_j can be expressed as a linear combination of the basis functions of V_{j+1} in the form as

$$\phi(x) = \sum_k h(k)\sqrt{2}\phi(2x - k) \tag{2}$$

where coefficient $h(k)$ is defined as $\langle \phi(x), \sqrt{2}\phi(2x - k)\rangle$.

Consider the orthogonal complement W_j of V_j to V_{j+1}, that is $V_{j+1} = V_j \oplus W_j$ with $\oplus$ being an operation of union. From this complement feature and $V_j \subset V_{j+1}$, it has the property of $V_{j+1} = V_0 \oplus W_0 \oplus W_1 \oplus \cdots \oplus W_j$. Define

$$\psi(x) = \sqrt{2}\sum_k (-1)^k h(-k+1)\phi(2x - k). \tag{3}$$

It can be shown that $\{\sqrt{2}\psi(2x - k); k \in Z\}$ is an orthonormal basis for W_j. The space W_j contains the detailed information needed to go from an approximation at resolution j to an approximation at resolution $j+1$. The family $\{\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k), j, k \in Z\}$ is a wavelet basis family for space $L^2(R)$.

With the chosen scaling function and the family of wavelet basis, a given function $f(t) \in L^2(R)$ can be decomposed on M levels. Suppose $g_i \in W_i$ and $f_i \in V_i$, the decomposition procedure yields

$$\begin{aligned} f(t) &= f_M(t) + \sum_{m=1}^{M} g_m(t) \\ &= \sum_k \lambda_M(k)\phi(2^M t - k) \\ &\quad + \sum_{m=1}^{M}\sum_k \gamma_m(k)\psi(2^m t - k) \end{aligned} \tag{4}$$

where $\lambda_M(k)$ and $\gamma_m(k)$ are the coefficients of decomposition.

When the wavelet packet algorithm is used, the original signal is firstly filtered by a half banded highpass filter and a half banded lowpass filter. After that, this procedure is repeated to the filtered two signals. Finally, a series of signals at different frequency bands can be obtained. From this process, we can see that if a signal is decomposed on M levels, we will obtain a series of signals on 2^M different frequency bands. This series of signals contains both frequency information and time information from which the error components in different frequency regions at different time steps can be identified. The original signal can be recovered from this series of signals.

More information about wavelet transform and wavelet packet algorithm can be found in [21]–[23].

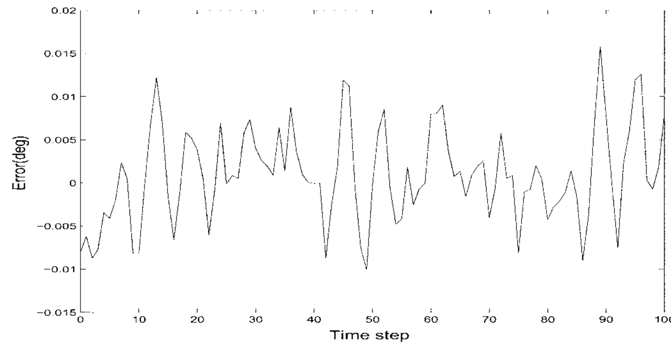

Fig. 1. Error signal at the 100th cycle.

B. Error Analysis Using Wavelet Packet Algorithm

To illustrate the usage of wavelet packet algorithm in our method, an example is provided. The error signal e_j is from an experiment at $j = 100$th cycle. After preprocessing to eliminate unwanted high-frequency components, the signal becomes $\tilde{e}_j$ and is shown in Fig. 1.

This error signal $\tilde{e}_j$ is decomposed by the wavelet packet algorithm and its decomposition result is a series of 2^M signals on different frequency regions. This series of signals is denoted as $\tilde{e}_j^i$ with j being the cycle index and $i \in [1, 2^M]$ being the index of frequency region. In this example, the error signal $\tilde{e}_j$ is decomposed on three levels (M $= 3$). The frequency range $[0, f]$, which is the frequency bandwidth of signal $\tilde{e}_j$, is evenly divided into $2^3 = 8$ frequency regions. Region 1 stands for the lowest frequency and region 8 the highest. The wavelet transform decomposes a signal with a component distribution over these regions and the decomposed error signal series $\tilde{e}_j^i$ is plotted in Fig. 2. The three axes of the coordinate are time step, magnitude, and frequency region index. At any one time step $k \in [1, p]$ with p being the total length of the trajectory, the maximal frequency component of the decomposed signal series at this time step $\tilde{e}_j^{m(j,k)} = \max_{i \in [1, 2^M]} \tilde{e}_j^i(k)$ can be located at any region. Furthermore, the region $m(j, k)$ that contains the maximal frequency components is termed as the distribution index of this time step. That is, the distribution index $m(j, k)$ is referred to the region that contains the maximal error component at the kth step of the jth cycle. It changes not only with time step, but also with operation cycle. For this example, the distribution index for this cycle is illustrated in Fig. 3.

From Fig. 3, it is clear that the distribution index at different time steps falls into different frequency regions. To show it clearly, the frequency components at three time steps are

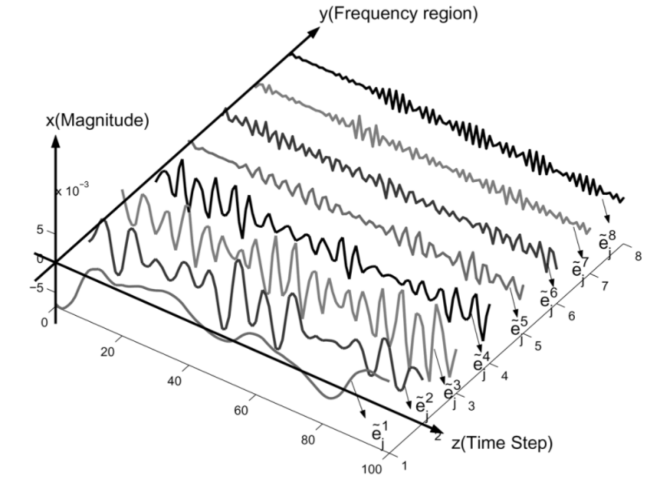

Fig. 2. Wavelet decomposition of error signal.

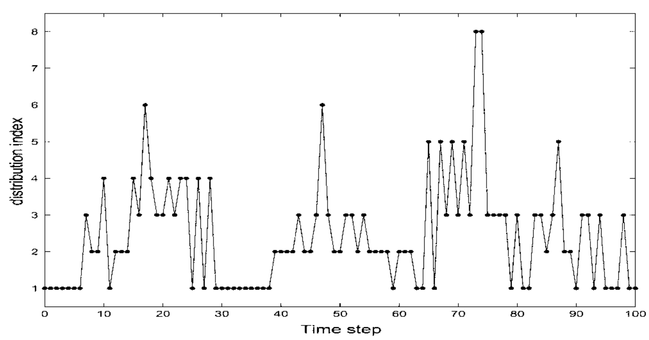

Fig. 3. Distribution index of maximal error component.

shown in Fig. 4. From this figure, we can see the maximal error component at the first time step is in the lowest frequency region $[0, (f/8)]$, i.e., the distribution index is in region 1. At time step 10, the maximal error component locates in the fourth frequency region $[(3f/8), (4f/8)]$, i.e., the distribution index is in region 4. At time step 74, the distribution index falls in the highest frequency region, $[(7f/8), f]$, i.e., the distribution index is in region 8.

Based on this distribution index, we can design a time-varying tuning filter $F_j(k)$ to filter the error signal of ILC system at the kth time step of the jth cycle. The cutoff frequency, denoted as $f_j(k)$, of the filter $F_j(k)$, is the upper bound of the distribution index at the kth time step. Hence, the filtered error signal contains the main error component at any one time step. For the example above, when we filter the error signal, the cutoff frequency of the filter $f_j(k)$ should be $f_j(1) = (f/8)$ at step 1, $f_j(10) = (4f/8)$ at step 10, and $f_j(74) = f$ at step 74. With such a tuning filter $F_j(k)$, all frequency components below $f_j(k)$, which is determined by the distribution index $m(j,k)$, are allowed to pass the filter. The design of the filter $F_j(k)$ will be discussed later.

Through this example, we can see that by using the wavelet packet algorithm, the frequency distribution index $m(j,k)$ at each time step can be identified. This distribution index will be used to determine the cutoff frequency of the tuning filter $F_j(k)$ at the corresponding time step. Based on this index from the

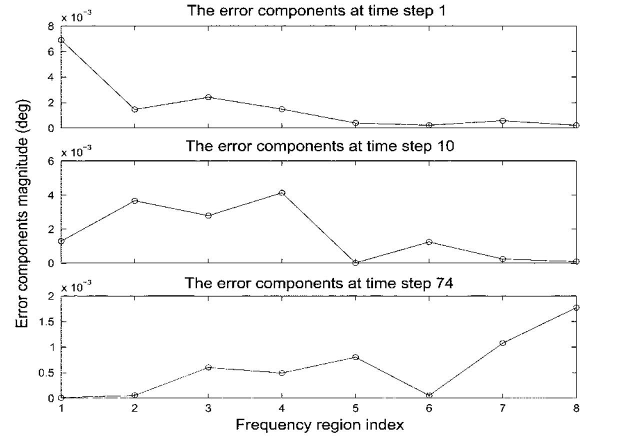

Fig. 4. Frequency components at different time steps.

wavelet transform, we propose a cutoff frequency tuning ILC in the following section.

III. Cutoff Frequency Tuning ILC

A trajectory may contain different frequency components at different time steps. For example, if the trajectory contains a sharp turn, the signal near the turning point contains many high-frequency components and it is desirable to let this information enter the learning for a better performance. On the other hand, for those points only containing low frequency components, a low cutoff is suitable for better learning transient and long-term stability. According to the distribution index $m(j,k)$ at each time step, an index dependent filter can be used.

Longman [9] suggested that it would be easy to implement if ILC adjusts the command given to the feedback control system. In this case, the existing feedback controller can be kept untouched. This approach is mathematically equivalent to adjust torque in ILC [24]. In this paper, this approach to adjust command is employed and the ILC update law with linear phase lead [8], [25], [26] will be used to highlight the advantage of the proposed method. The update law is written as

$$\begin{cases} u_j(k) = y_d(k) + u_{L,j}(k) \\ u_{L,j+1}(k) = u_{L,j}(k) + \gamma e_j(k+l) \end{cases} \tag{5}$$

where j is cycle index, k is time step, γ is learning gain, and l is lead-step. $e_j(k) = y_d(k) - y_j(k)$ is the error signal at the jth cycle, in which $y_d(k)$ is the desired trajectory and $y_j(k)$ is the actual trajectory at the jth cycle. $u_{L,j}$ is the adjustment of command in the jth cycle and u_j is the input to the closed-loop feedback control system.

With this update law, Longman *et al.* [6], [9], [10], [24], [26] provided the discrete frequency domain condition of monotonic decay of error for the time-invariant linear system as follows

$$|1 - \gamma z^l G(z)| < 1; z = e^{j\omega} \quad \text{with} \quad \omega \in [0, \omega_n] \tag{6}$$

where $G(z)$ is system model, ω_n is the Nyquist frequency. Longman *et al.* pointed out the difficulties to make this condition hold for all frequencies [8]. All such frequencies that make this condition hold form a learnable band. The upper-limit of this band is called the learnable bandwidth. To guarantee good learning transient, the frequency components entering the learning should be in this learnable band. A simple way to realize this goal is using a zero-phase low-pass filter.

In this paper, a cutoff frequency tuning method is proposed with the feature of time-varying cutoff frequency as follows

$$\begin{cases} u_j(k) = y_d(k) + u_{L,j}(k) \\ u_{L,j+1}(k) = u_{L,j}(k) + \gamma F_j(k) e_j(k+l) \\ \qquad = u_{L,j}(k) + \gamma e_j^*(k+l) \end{cases} \tag{7}$$

where $F_j(k)$ is the filter at time step k of operation cycle j and $e_j^*(k+l) = F_j(k)e_j(k+l)$ is error signal after filtering.

A. *Cutoff Frequency Tuning Scheme*

In the proposed method, the error signal e_j needs to be preprocessed by eliminating noises, unmodeled uncertainties, and unwanted high-frequency components above an estimated learnable bandwidth f_b. The value f_b can be obtained from system model. The preprocessed error signal $\tilde{e}_j$ is decomposed by wavelet packet algorithm and the distribution index $m(j,k)$ at any one time step can be identified. At any one time step during an operation cycle, the cutoff frequency of the filter $F_j(k)$ is set based on the distribution index. Signal $\tilde{e}_j$ is filtered by the time-varying tuning filter $F_j(k)$ with cutoff frequency of $f_j(k)$ and the filtered signal is used to update the input signal as in (7). In our description, the time-varying filter means at each time step k, the filter $F_j(k)$ has a different cutoff frequency.

The scheme of this cutoff frequency tuning ILC is illustrated in Fig. 5. In this figure, $\mathbf{C}$ is a conventional feedback controller and $\mathbf{P}$ is a plant. They form a closed-loop feedback control system. From this figure, the implementation of the cutoff frequency tuning ILC can be summarized as follows.

1) Preprocess the error signal e_j. This yields $\tilde{e}_j$.
2) Decompose $\tilde{e}_j$ and we get a series of 2^M signals on different frequency regions. This series of signals is denoted as $\tilde{e}_j^i(k)$ with $i \in [1, 2^M]$ being the index of frequency region, $j \in [1,\infty]$ being the cycle index, and $k \in [1,p]$ the index of time step with p being the total length of trajectory.
3) For each time step k, define the distribution index $m(j,k) \in [1,2^M]$ such that $\tilde{e}_j^{m(j,k)}(k) = \max_{i\in[1,2^M]} \tilde{e}_j^i(k)$.
4) For each time step k, set the cutoff frequency of tuning filter $F_j(k)$ as $f_j(k) = (m(j,k))/(2^M) \cdot f_b$. That is, the cutoff frequency is the upper bound of the frequency region where the maximal error component resides.
5) Use the filter $F_j(k)$ with time varying cutoff frequency $f_j(k)$ to filter $\tilde{e}_j$. Then, add lead-step l to yield the signal e_j^*. This signal is used in (7) to update the input signal.
6) Execute next operation cycle, record the error signal e_{j+1} and return to step 1.

B. *Design of Zero-Phase Low-Pass Filter $F_j(k)$*

To simplify the computation of zero-phase low-pass filter $F_j(k)$, a window filter is used. For filter $F_j(k)$ with cutoff

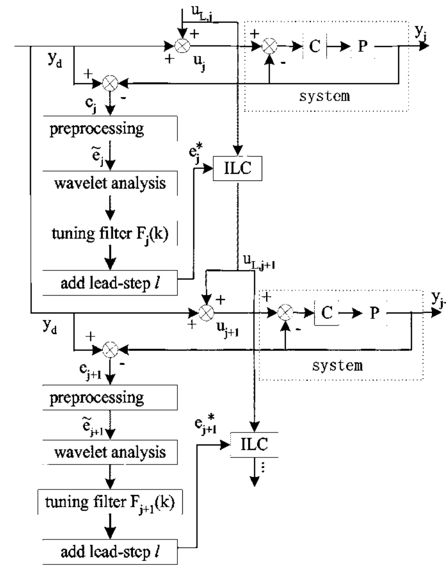

Fig. 5. Scheme of frequency tuning iterative learning control.

frequency of $f_j(k)$ rad/s, its impulse response sequence $z_j^k(n)$ can be obtained from its frequency response $H_{j,k}(\omega)$ [27]

$$\begin{aligned} z_j^k(n) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} H_{j,k}(\omega)e^{j\omega n}\,d\omega \\ &= \frac{f_j(k)}{\pi}\operatorname{sinc}\left(\frac{f_j(k)}{\pi}n\right). \end{aligned} \tag{8}$$

The generated $z_j^k(n)$ is not implementable in practice because impulse response $z_j^k(n)$ is infinite. To create finite-duration impulse response, a hamming window is employed to truncate the infinite impulse response $z_j^k(n)$. This hamming window is defined as [27]

$$w_j^k(h) = \begin{cases} 0.54 - 0.46\cdot\cos\frac{2h\pi}{N-1} & h\in[0,N-1] \\ 0 & \text{otherwise} \end{cases}.$$

where N is the width of Hamming window. In our ILC learning system, this N corresponds to N sampling points. Finally, the impulse response of the filter $F_j(k)$ is obtained as

$$\hat{z}_j^k(h) = z_j^k(n)\cdot w_j^k(h). \tag{9}$$

The generated $z_j^k(h)$ with $h \in [0, N-1]$ is the weighting factor of each sampling point in the window.

For a window filter, the filtering point is placed at the middle of the window to realize zero-phase. With this filter, the learning law in (7) can be written as

$$\begin{cases} u_j(k) = y_d(k) + u_{L,j}(k) \\ u_{L,j+1}(k) = u_{L,j}(k) \\ \quad +\gamma\sum_{h=0}^{N-1}\hat{z}_j^k(h)e_j\left((k+l)+\left(h-\frac{N-1}{2}\right)\right) \end{cases} \tag{10}$$

in which $e_j((k+l)+(h-(N-1)/(2)))$ is the sampling point of the error signal corresponding to weighting factor $\hat{z}_j^k(h)$ with $h\in[0,N-1]$.

Written this in matrix form, we have

$$\begin{cases} U_j = Y_d + U_{L,j} \\ U_{L,j+1} = U_{L,j} + \gamma\hat{Z}_j^k E_j \end{cases} \tag{11}$$

with $U_j = [u_j(0), u_j(1), \ldots, u_j(p-1)]^T, Y_d = [y_d(1), y_d(2), \ldots, y_d(p)]^T$

$$\hat{Z}^k = \begin{bmatrix} \hat{z}^1(0) & \hat{z}^1(1) & \cdots & \hat{z}^1(N-1) \\ \hat{z}^2(0) & \hat{z}^2(1) & \cdots & \hat{z}^2(N-1) \\ \vdots & \vdots & \ddots & \vdots \\ \hat{z}^p(0) & \hat{z}^p(1) & \cdots & \hat{z}^p(N-1) \end{bmatrix}$$

$$E_j = \begin{bmatrix} e_j(l-m) & \cdots & e_j(l-m+p-1) \\ e_j(l-m+1) & \cdots & e_j(l-m+p) \\ \vdots & \ddots & \vdots \\ e_j(l+m) & \cdots & e_j(l+m+p-1) \end{bmatrix}$$

in which $m = (N-1)/(2), e_j(\Delta) = e_j(1)$ for $\Delta < 1$, and $e_j(\Delta) = e_j(p)$ for $\Delta > p$.

Remark 1: To realize zero-phase filtering and minimize the influence of initial state, the error signal is extended on both ends [28]. For computation simplicity, the error signal $\tilde{e}_j$ is extended by repeating the end-points of the signal and these added points are cut after the filtering to get the filtered signal.

Compared with previous works, our filter design is simple. Chen's method [11] uses a B-spline network to build the filter. The designed filter "is close to zero-phase filter in low frequencies" and "phase distortion at high frequencies may go up to $\pm 90°$ [11]." Hence, the learning performance will be attenuated. Zheng's method [29] uses a Q-filter. The relationship between filter parameters and bandwidth need to be estimated and more design work is needed.

IV. Experiments

In this section, some experimental results are given to verify the proposed cutoff frequency tuning scheme. The experiment is carried out on a joint moving in the horizontal plane of an industrial robot, SEIKO TT3000, which is a SCARA type robotic manipulator with four joints. Its sampling period is 0.01 second. Hence, its Nyquist frequency is 50 Hz.

In the experiments, the lead-step l is set as 5. Wirkander *et al.* pointed out that learning gain has little influence on performance [8] and Longman *et al.* suggested the learning gain should be a low value [30]. Hence, the learning gain γ is set as 1. The learning performance of the proposed cutoff frequency tuning ILC and that of a conventional fixed filter ILC will be compared. For both methods, the window filter discussed in Section III-B is used.

Before the experiments, the learnable bandwidth of the learning system needs to be estimated. A rough system model is identified for this purpose as follows:

$$G(z) = \frac{0.022\,77z}{z^2 - 1.659z + 0.683}. \qquad (12)$$

With $\gamma = 1, l = 5$, and system model (12), the learnable bandwidth can be obtained from (6). This condition is illustrated in Fig. 6. From this figure, the learnable bandwidth is approximately read as 13.7 Hz.

The desired trajectory is specified in joint space and contains a smooth path for an about $10°$ turn followed by a return to the starting point in 1 second. This trajectory contains only one frequency component, which is a normal cosine wave, and is shown in Fig. 7. All the following experimental results are based on this trajectory if no special statement is made.

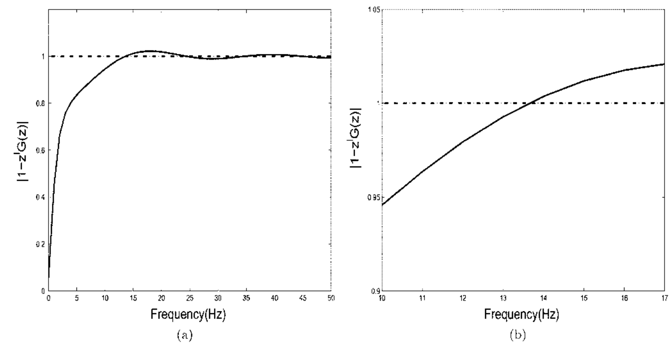

Fig. 6. Bode magnitude plot of $|1 - z^l G(z)|$. (a) Bode magnitude. (b) Zoomed Bode.

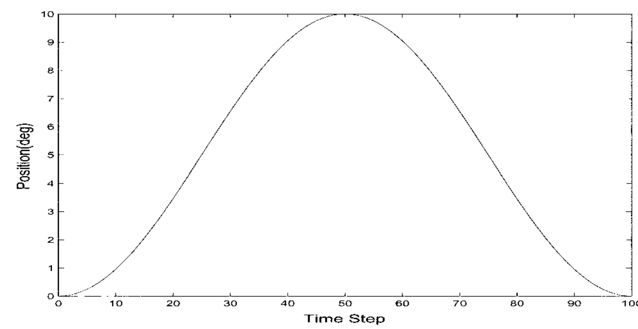

Fig. 7. Trajectory with uniform frequency.

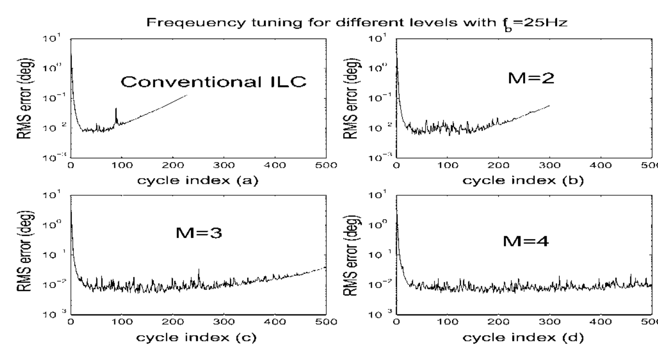

Fig. 8. Influence of decomposition level.

A. Determination of the Decomposition Level

In our proposed method, the discrete wavelet transform is used to make computation efficient. A parameter M, the level of decomposition, needs to be determined to decompose the error signal on 2^M frequency regions.

If M is too small, the adjustment of cutoff frequency is coarse and the beneficial effect of the cutoff frequency tuning scheme is not obvious. On the contrary, a large M can get a fine tuning of cutoff frequency but the tradeoff is more computation time. Thus, it is not advisable to set M at too high a value.

To see the influence of the decomposition level M, Fig. 8 shows the experimental results based on an ILC with learning gain $\gamma = 1$, lead-step $l = 5$, and an estimated learnable bandwidth $f_b = 25$ Hz. From Fig. 6, we know the learnable bandwidth is 13.7 Hz, which is much lower than this estimation of

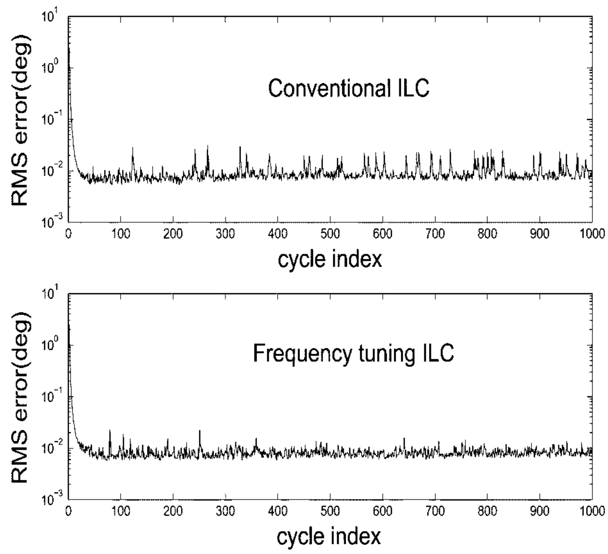

Fig. 9. RMS error of lead-step 5 and cutoff 15 Hz.

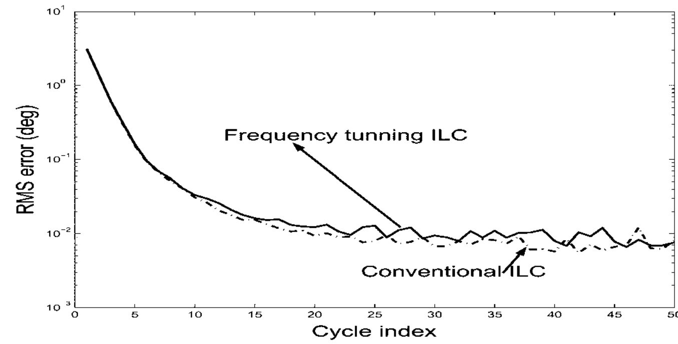

Fig. 10. RMS error at the first 50 cycles.

25 Hz. Hence, the learning for conventional ILC with fixed filter diverges at about the 50th cycle, which can be seen in Fig. 8(a). When level $M = 2$, the coarse adjustment leads learning to divergence at about the 150th cycle in Fig. 8(b). A level $M = 3$ can reduce this divergent trend drastically in Fig. 8(c). When level is set as 4, there is no divergence trend in the first 500 cycles as shown in Fig. 8(d). This indicates that the cutoff frequency tuning method with a large M, which implies a fine adjustment of cutoff frequency, works well. Hence, the level will be set as 3 or 4 in following applications. The level 2 is not used because of its poor performance.

B. Experimental Results

In this section, two experimental results are presented. The first one is the comparison between our cutoff frequency tuning ILC and conventional ILC with estimated learnable bandwidth equals to the actual learnable bandwidth. This learnable bandwidth will yield the best learning performance for conventional ILC. The second one is for a trajectory contains more frequency components with estimated learnable bandwidth higher than the actual learnable bandwidth to show that the proposed method can deal with this situation.

Experiment 1: Since the model is inaccurate, the estimated learnable bandwidth f_b is set as 15 Hz, which is different from the value of 13.7 Hz we got from Fig. 6. 15 Hz is the actual learnable bandwidth and gives the best learning performance for conventional fixed filter ILC. We must point out that the actual learnable bandwidth of a system is often unknown. Here, the actual learnable bandwidth 15 Hz is obtained from many experiments for comparison purpose. For cutoff frequency tuning ILC, the level of decomposition is 3. The results are shown in Fig. 9.

The advantage of our cutoff frequency tuning scheme is not obvious in this experiment. But some advantages can be obtained when the results are carefully compared. After learning has reached steady state, both methods produce comparable accuracy with the proposed method achieving about 10% better than conventional ILC. In addition, from the root square mean (RMS) error of conventional ILC, it is clear to see that there are many peaks, which means that this conventional ILC suffers from the high-frequency noises and uncertainties.

Let us see the RMS error in the first 50 cycles in Fig. 10. The conventional ILC has a convergence speed a bit faster than cutoff frequency tuning scheme in the first 50 cycles. At the early cycles, the main error components stay in low frequencies and the cutoff frequency of filters at each step in our method often be low. In this case, when cutoff frequency tuning filter ILC is used, some error components in high frequencies do not enter the learning in these cycles and this causes the learning speed of cutoff frequency tuning scheme in these cycles a bit slow while the conventional ILC does not have this problem. But we can see from the figure that this has only very little influence on the performance.

This experiment shows that the proposed method has advantage over conventional ILC. We also did an experiment for a higher f_b, which is omitted here. When the estimated learnable bandwidth f_b is set as 17 Hz, the experimental results show that conventional ILC leads a very quick divergent learning behavior while the proposed method has a monotonic decay of error.

Experiment 2: This experiment investigates a f_b higher than the actual learnable bandwidth for a trajectory contain more frequency components. In practice, many applications have desired trajectories with wide range of frequency components. The frequency components of the trajectory at different parts vary and the proposed cutoff frequency tuning ILC method should be able to adapt to the situation. In this experiment, the desired trajectory is given as follows and is illustrated in Fig. 11.

$$y_d(i) = \begin{cases} a \times ((i-1)^2/2): & i \in [1, 30] \\ b \times (i-16): & i \in [31, 47] \\ \begin{pmatrix} (c \times (i-48.5)^2)/2 \\ +b \times (i-48.5) + d \end{pmatrix}: & i \in [48, 52] \\ 11 - b \times (i-51): & i \in [53, 70] \\ \begin{pmatrix} (a \times (i-71)^2)/2 \\ -b \times (i-71) + e \end{pmatrix}: & i \in [71\,100] \end{cases}$$

in which i is the index of sampling point, $a = 0.010\,476\,19, b = 0.314\,285\,714, c = -0.125\,714\,28, d = 10.214\,285\,71$, and $e = 4.714\,285\,714$.

In this experiment, the lead-step $l = 5$ and learning gain $\gamma = 1$. The estimated learnable bandwidth is $f_b = 17$Hz and the decomposition level is set as 4. The experimental results are shown in Fig. 12. We can see the RMS error of conventional ILC with fixed filter shows a very poor learning transient. It diverges

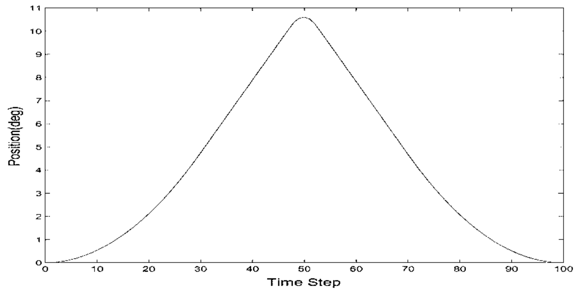

Fig. 11. Trajectory with different frequencies.

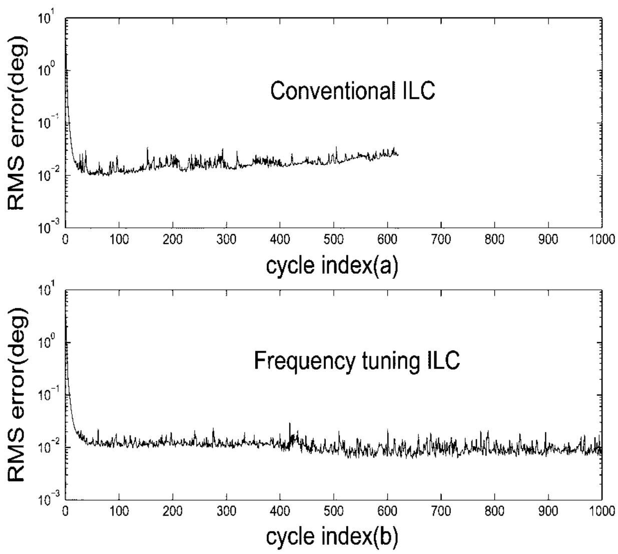

Fig. 12. RMS error of lead-step 5 and cutoff 17 Hz.

from about the 100th cycle and makes some noise at about the 600th cycle so that we have to stop the experiment. On the contrary, the tuning scheme shows a good learning transient and good tracking error. The RMS error remains stable and it continuously goes down after about 500 cycles. The tracking error in the first 500 cycles reaches 0.012° while the tracking error in the last 500 cycles reaches 0.0091°. The tracking performance is further improved. The reason of this can be explained as follows: after about 500 cycles, the main error components begin to move into the frequency around 17 Hz. The error components in this frequency become the main error components and they begin to enter the learning to further improve the performance so that the error level can be further improved.

The power spectrum of the error signal for both our proposed method and conventional ILC are shown in Fig. 13. It is clear that the power spectrum of error signal for cutoff frequency tuning ILC is much less than the that of error for conventional ILC, especially in the frequency region [13 Hz, 17 Hz].

The input signals of different schemes are shown in Fig. 14. It can be seen that the input signal of the conventional ILC has become oscillatory with very big high-frequency components, while that of the cutoff frequency tuning scheme keeps smooth. This experiment shows that this cutoff frequency tuning scheme can deal with the trajectory with different frequency components with a higher estimated learnable bandwidth.

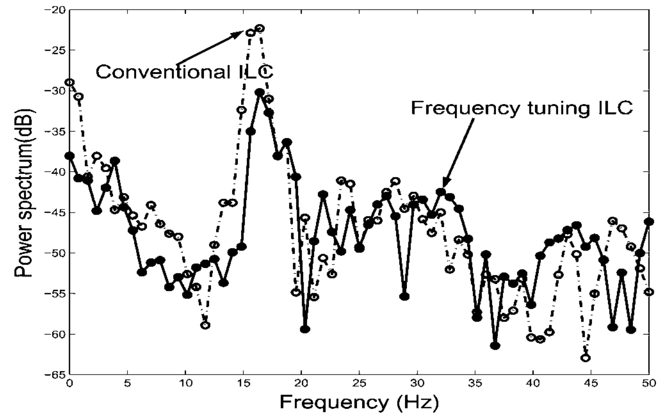

Fig. 13. Power spectrum comparison.

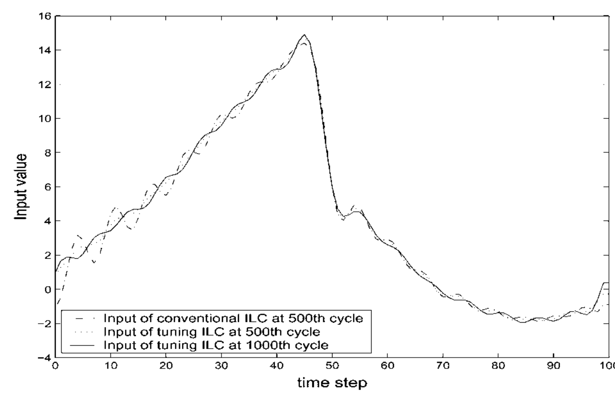

Fig. 14. Input signals of lead-step 5 and cutoff 17 Hz.

From these experiments, we can see that cutoff frequency tuning ILC work well for a properly enlarged learnable bandwidth. Because the system model is often inaccurate, the estimated learnable bandwidth f_b obtained from condition (6) is not likely to match the actual learnable bandwidth. To guarantee good learning behavior, f_b is often chosen as a conservative value and this will degrade the tracking performance. While in our method, f_b can be chosen in a broader region and learning performance can be guaranteed. This is very desirable in practice.

V. Conclusion

In this paper, a cutoff frequency tuning method based on time-frequency analysis of error signal at each cycle is proposed and some experimental results are provided to verify the method. In this method, the cutoff frequency of the filter is a function of time as well as the index of cycle. From experiment results, it can be seen that the proposed method works well. This cutoff frequency tuning scheme outperforms its conventional ILC counterpart in that: firstly, this cutoff frequency tuning scheme can let high-frequency information enter learning at proper time steps and can minimize the unwanted high-frequency components by using a filter with a cutoff frequency that covers only the major error components so that the learning transient and long-term stability can be improved. Secondly, the proposed cutoff frequency tuning method allows the estimated learnable bandwidth in a broader region. Experimental results show that the proposed cutoff frequency tuning scheme can work quite well for a cutoff frequency where conventional ILC will diverge very quickly.

REFERENCES

[1] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *J. Robot. Syst.*, vol. 1, pp. 123–140, 1984.

[2] G. Casalino and G. Bartolini, "A learning procedure for the control of movements of robotic manipulators," in *Proc. 4th IASTED Symp. Robotics Automation*, Amsterdam, The Netherlands, 1984, pp. 108–111.

[3] J. J. Craig, "Adaptive control of manipulator through repeated trials," in *Proc. American Control Conf.*, San Diego, CA, 1984, pp. 1566–1573.

[4] R. H. Middleton, G. C. Goodwin, and R. W. Longman, "A method for improving the dynamic accuracy of a robot performing a repetitive task," *Int. J. Robot. Res*, vol. 8, no. 5, pp. 67–74, 1989.

[5] C. J. Goh, "A frequency domain analysis of learning control," *J. Dynamic Syst., Measurement, Contr.*, vol. 116, pp. 781–786, Dec. 1994.

[6] C.-K. Chang, R. W. Longman, and M. Q. Phan, "Techniques for improving transients in learning control systems," *Adv. Astronautic. Sci.*, vol. 76, pp. 2035–2052, 1992.

[7] H.-S. Lee and Z. Bien, "A note on convergence property of iterative learning controller with respect to sup norm," *Automatica*, vol. 33, no. 8, pp. 1591–1593, 1997.

[8] S.-L. Wirkander and R. W. Longman, "Limit cycles for improved performance in self-tuning learning control," *Adv. Astronautic. Sci.*, vol. 102, pp. 763–781, 1999.

[9] R. W. Longman, "Iterative learning control and repetitive control for engineering practice," *Int. J. Contr.*, vol. 73, no. 10, pp. 930–954, 2000.

[10] H. Elci, R. W. Longman, M. Phan, J.-N. Juang, and R. Ugoletti, "Simple learning control made practical by zero-phase filtering: Application to robotics," *IEEE Trans. Circuit Syst. 1, Fundam. Theory Appl.*, vol. 49, pp. 753–767, Jun. 2002.

[11] Y.-Q. Chen and K. L. Moore, "Frequency domain adaptive learning feedforward control," in *Proc. IEEE Symp. Computational Intelligence Robotics Automation*, Banff, AB, Canada, Jun.–Aug. 2001, pp. 396–401.

[12] D. Wang and Y. Ye, "Analysis and design of anticipatory learning control," in *Proc. 42nd Conf. Decision Control*, Dec. 2003, pp. 4434–4439.

[13] R. W. Longman and T. Kwon, "Obtaining good transients in iterative learning control using step response data," in *Proc. 2002 AIAA/AAS Astrodynamics Specialist Conf. Exhibit*, Monterey, CA, Aug. 2002, pp. 1–10.

[14] X. Tang, L. Cai, and W. Huang, "A learning controller for robot manipulators using Fourier series," *IEEE Trans. Robot. Automat.*, vol. 16, no. 1, pp. 36–45, Feb. 2000.

[15] B. Zhang, D. Wang, and Y. Ye, "Cutoff-frequency phase-in method to improve tracking accuracy," in *Proc. 5th Asian Control Conf.*, Australia, Jul. 2004, pp. 930–935.

[16] M. Norrlöf, "Iteration varying filters in iterative learning control," in *Proc. 4th Asian Control Conf.*, Singapore, 2002, pp. 2124–2129.

[17] D.-N. Zheng and A. Alleyne, "Stability of a novel iterative learning control scheme with adaptive filtering," in *Proc. American Control Conf.*, New Orleans, LA, Jun. 2003, pp. 4512–4517.

[18] I. Rotariu, R. Ellenbroek, and M. Steinbuch, "Time-frequency analysis of a motion system with learning control," in *Proc. Amercian Control Conf.*, Denver, CO, Jun. 2003, pp. 3650–3654.

[19] B. Zhang, D. Wang, and Y. Ye, "Experimental study of time-frequency based ILC," in *Proc. 8th Int. Conf. Control, Automation, Robotics, Vision*, Kunming, China, Dec. 2004.

[20] J.-X. Xu and Y. Tan, *Linear and Nonlinear Iterative Learning Control*, ser. Series of Lecture Notes in Control and Information Sciences 291. Berlin, Germany: Springer-Verlag, 2003.

[21] C. Valens. (1999) A Really Friendly Guide to Wavelets. [Online]. Available: http://perso.wanadoo.fr/polyvalens/clemens/wavelets/wavelets.html

[22] R. Polikar. The Wavelet Tutorial. [Online]. Available: http://www.public.iastate.edu/rpolikar/Wavelets

[23] "Wavelet Toolbox—User's Guide,", 1997. The MathWorks Inc..

[24] E. J. Solcz and R. W. Longman, "Disturbance rejection in repetitive controller," *Advances Astronautical Sciences*, vol. 76, pp. 2111–2130, 1992.

[25] D. Wang, "On D-type and P-type ILC designs and anticipatory approach," *Int. J. Contr.*, vol. 73, pp. 890–901, 2000.

[26] Y. Wang and R. W. Longman, "Use of noncausal digital signal processing in learning and repetitive control," *Adv. Astronautic. Sci.*, vol. 90, pp. 649–668, 1996.

[27] "Signal Processing Toolbox—User's Guide," MathWorks, Inc., Natick, MA, 1997.

[28] A. M. Plotnik and R. W. Longman, "Subtleties in the use of zero-phase low-pass filtering and cliff filtering in learning control," *Advances Astronautical Sciences*, vol. 103, pp. 673–692, 1999.

[29] D.-N. Zheng, "Iterative learning control of an electro-hydraulic injection molding machine with smoothed fill-to-pack transition and adaptive filtering," Ph.D. dissertation, Dept. Mechanical Eng., Univ. Illinois, Urbana-Champaign, IL, 2002.

[30] R. W. Longman and S.-L. Wirkander, "Automated tuning concepts for iterative learning and repetitive control laws," in *Proc. 37th Conf. Decision Control*, Tampa, FL, 1998, pp. 192–198.

Bin Zhang (S'00) received the B.E. and M.S.E. degrees in control engineering from Nanjing University of Science and Technology, Nanjing, China, in 1993 and 1999, respectively. He is currently working toward the Ph.D. degree at Nanyang Technological University, Nanyang, Singapore.

His current research interests are in the fields of control systems and theory, iterative learning control, repetitive control, intelligent control, and their applications to robot manipulators and power electronics.

Danwei Wang (M'00) received the B.E. degree from the South China University of Technology, Guangzhou, China, in 1982 and the Ph.D. and M.S.E. degrees from the University of Michigan, Ann Arbor, in 1989 and 1984, respectively.

Since 1989, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang, Singapore. Currently, he is an Associate Professor and Director of the Center for Intelligent Machines, NTU. He has served as General Chairman, Technical Chairman, and various other positions in international conferences, such as International Conference on Control, Automation, Robotics and Vision (CARCVs) and Asian Conference on Computer Vision (ACCV). His research interests include robotics, control theory, and applications. He has published more than 150 technical articles in the areas of iterative learning control, repetitive control, robust control and adaptive control systems, as well as manipulator/mobile robot dynamics, path planning, and control.

Dr. Wang is an Associate Editor of Conference Editorial Board, IEEE Control Systems Society, and an active member of IEEE Singapore Robotics and Automation Chapter. In 1996 and 1997, he was a recipient of Alexander von Humboldt fellowship, Germany.

Yongqiang Ye received the B.E. and M.S. degrees from Zhejiang University, Zhejiang, China, in 1994 and 1997, respectively, and the Ph.D. degree from Nanyang Technological University, Nanyang, Nanyang, Singapore, in 2004, all in electrical engineering.

From June 1997 to September 2000, he was an Assistant Teacher and then a Lecturer with the Department of Information, Zhejiang Institute of Finance and Economics, China. Currently, he is with the School of Information, Zhejiang Institute of Finance and Economics, China. His research focuses on the areas of iterative learning control and repetitive control with applications to manipulator and power electronics. He has authored or coauthored over 20 journal and conference papers in the area of iterative learning control, repetitive control, and applications.