Control Engineering Practice 26 (2014) 116–124

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Case studies of filtering techniques in multirate iterative learning control

Bin Zhang [a,*], Yongqiang Ye [b], Keliang Zhou [c], Danwei Wang [d]

[a] Department of Electrical Engineering, University of South Carolina, Columbia SC, USA
[b] College of Automation Engineering, Nanhang University, Nanjing, Jiangsu, China
[c] Department of Electrical and Computer Engineering, University of Canterbury, New Zealand
[d] School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore

ARTICLE INFO

Article history:
Received 2 May 2013
Accepted 17 January 2014
Available online 11 February 2014

Keywords:
Iterative learning control
Multirate processing
Anti-aliasing
Anti-imaging

ABSTRACT

Iterative learning control (ILC) is a simple and efficient solution to improve tracking accuracy for systems that execute repetitively the same tracing operation. For engineering applications of ILC, the main concern is the monotonic decay of tracking errors, in the sense of infinity norm or peak error, along the trials. Low cost in implementation and robustness in performance are also critical factors. To achieve these important but sometimes contradicting goals, several multirate ILC schemes have been developed, in which different data sampling rates are used for feedback online loop and feedforward ILC offline loop. That is, multirate ILC uses a different (often lower) rate from the sampling rate of a feedback system to update input. Before the input signal is applied to the system for the next trial, it is upsampled to reach the original sampling rate. Since downsampling will cause distortion of frequency spectra, anti-aliasing and anti-imaging filters and signal extension are used together with downsampling and upsampling operations. In this paper, these technologies are integrated with three different multirate ILC schemes, pseudo-downsampled ILC, two-mode ILC, and cyclic pseudo-downsampled ILC, to achieve better performance. A series of experimental results on an industrial robot are presented to demonstrate the efficiency of multirate ILC schemes and compare the performance. The results demonstrate that multirate ILC schemes are able to achieve not only monotonic learning transient, but also much better tracking accuracy than conventional one-step-ahead ILC schemes.

1. Introduction

Iterative Learning Control (ILC) is an approach to find appropriate control input for systems that execute the same tracking task repeatedly. It aims to force the output of these systems to follow a trajectory $y_d(t)$ defined over a finite time duration T. With technology development, tracking accuracy requirements have come to nano- or micro-meter level. Feedback control alone is often not enough to achieve accuracy at this level due to modeling uncertainties and various disturbances. ILC provides a simple and effective feedforward channel to significantly improve the tracking accuracy with low cost.

The basic idea of ILC is to update the input through the recorded tracking error in a previous trial, or iteration. ILC is a batch processing process. After the execution of one trial, the input and error signals are recorded in the memory. Before the start of the next iteration, feedforward ILC controller offline updates the input signal. When the next iteration starts, the calculated input signal is applied to the system. The features of batch processing and off-line calculation enable ILC to employ techniques that cannot be used in real time, such as non-causal filtering.

The ILC update law has the general recursive form as

$$u_{j+1} = H(u_j, e_j) \tag{1}$$

where H is the ILC input update function; tracking error is $e_j = y_d - y_j$ with y_d and y_j being the desired trajectory and actual trajectory of the jth iteration, respectively. The objective is to make e_j converge to zero as iterations go to infinity. To describe e_j in a trial with a finite time duration or a finite number of sampling points, a certain norm $\|e_j\|$ is used. Therefore, ILC aims to achieve $\lim_{j\to\infty}\|e_j\| \to 0$. Note that ILC is a two-dimensional problem. On the one hand, the system performs the finite-time tracking command on the time axis. On the other hand, ILC adjusts the input to the system on the iteration axis. Time and iteration index are two independent variables (Elci, Longman, Phan, Juang, & Ugoletti, 2002).

* Corresponding author.
E-mail addresses: zhangbin@cec.sc.edu (B. Zhang), melvinye@nuaa.edu.cn (Y. Ye), eklzhou@ieee.org (K. Zhou), edwwang@ntu.edu.sg (D. Wang).

0967-0661/$ - see front matter
http://dx.doi.org/10.1016/j.conengprac.2014.01.018

Generally speaking, there are two configurations of the ILC scheme. The first one is parallel configuration (Bristow, Tharayil, & Alleyne, 2006; de Roover & Bosgra, 2000), in which ILC adjusts the commands to the plant directly. The second one is serial configuration (Longman, 2000), in which ILC adjusts the commands given to the existing closed-loop feedback control system. It has been proved that these two configurations are mathematically equivalent (Solcz & Longman, 1992). Since many commercial products have feedback controllers and it is not desirable to open the feedback control loops, the second configuration is relatively easier for practical implementations.

When an iterative learning controller is designed, an important issue that needs to be taken into account is the monotonic convergence of tracking error along the iteration axis (Chang, Longman, & Phan, 1992; Longman, 2000; Wang, 2000). As mentioned earlier, a proper norm is used to describe the tracking error in a trail such that the convergence is considered in the sense of this selected norm. It is well-known that ILC often shows bad transient behavior. That is, the tracking error goes down in the initial iterations but goes up again, usually to a very huge value, before it finally converges to zero. The reason is that many previous analyses are either in the α-norm or λ-norm. The λ-norm for a function $f(t)$ is $\|f\|_\lambda \triangleq \sup_{t\in[0,T]} e^{-\lambda t}\max|f(t)|$ with λ being a positive scalar that usually needs to be sufficiently large. The α-norm is defined as $\|f(\cdot)\|_\alpha = \sup_{k\in N} f(k)\alpha^k$ with $0<\alpha<1$. In the sense of these two norms, the error near the terminal phase of the operations is much less weighted than those at the beginning phase of the operations. Due to this decreasing weighting factor, a huge overshoot of error can appear and indicate a bad convergence performance in the sense of the ∞-norm, given by $\|f\|_\infty = \sup_{k\in N} f(k)$, even with the presence of a mathematical convergence analysis with α-norm or λ-norm. To overcome this bad transient behavior, the convergence should be investigated in the ∞-norm and many approaches have been developed (Frueh & Phan, 2000; Jang, Chio, & Ahn, 1995; Kuc, Lee, & Nam, 1992; Lee-Glauser, Juang, & Longman, 1996; Park & Bien, 2002; Ye, Wang, Zhang, & Wang, 2009; Zhang, Wang, & Ye, 2009; Zhang, Wang, & Ye, 2010).

The explanation in the frequency domain is that bad transient behavior is due to high frequency error components violating the condition of monotonic decay or the error signal contains a component beyond the ILC system's learnable bandwidth (Zhang, Wang, & Ye, 2005). A widely used method to achieve good learning behavior is to introduce a low-pass filter (Chen & Moore, 2001; Zhang, Wang, & Ye, 2009). However, ILC with such a filter will no longer be able to achieve zero tracking since it cuts off high frequency components. If desired performance requires elimination of error components in high frequencies, this method results in poor tracking accuracy.

It is desirable, therefore, to develop ILC to guarantee both transient behavior and high tracking accuracy in the form of infinite norm. In Moore, Chen, and Bahl (2005), Moore et al. derived an exponential convergence condition for P-type ILC and they used time-varying gain to make the condition hold. For most systems in use, the limitation is that the feedback controller is encapsulated and the condition from Moore et al. (2005) often cannot be satisfied. Redesign feedback controller to satisfy the condition is inconvenient (Moore, Chen, & Bahl, 2002). Alternatively, a simple solution to make the condition in Moore et al. (2005) hold is to reduce the sampling rate. Since it is not easy to change sampling frequency for most physical systems, multirate ILC schemes are developed in which the sampled data are processed at different rates. This will bring the design of ILC into the multirate signal processing domain (Zhang, Wang, Wang, Ye, & Zhou, 2008; Zhang, Wang, Ye, Wang, & Zhou, 2007, 2008; Zhang, Wang, Ye, Zhou, & Wang, 2010). Some other approaches using the multirate concept include optimal ILC for multirate physical systems and multirate model inverse (Oomen, Wijdeven, & Bosgra, 2009; Shiraishi & Fujimoto, 2010).

Another advantage of multirate ILC is that it is able to deal with initial state error properly. The original definition of ILC problem requires the same initial state of each iteration (Longman, 2000). This makes analysis simple and makes zero-error tracking possible. However, this assumption may not hold for real systems because the same initial state sometimes cannot be guaranteed in practice. A research on continuous D-type ILC shows that initial state error can make the learning process unstable (Lee & Bien, 1991). Some methods are proposed to achieve good learning behavior with the presence of initial state error (Chen, Wen, Gong, & Sun, 1999; Chen, Wen, Xun, & Sun, 1996; Hillenbrand & Pandit, 2000; Sun & Wang, 2002; Wang, 2000).

It is worth noting that multirate control itself is not new and its capabilities and limitations have been well-studied (Moore, Bhattacharyya, & Dahleh, 1993). One limitation is the degradation in the intersample behavior. This is also true for multirate ILC and we will design novel ILC schemes to overcome this limitation. Based on the work in Moore et al. (2005), several multirate ILC schemes are developed and successfully applied to an industrial robot system (Zhang, Wang, Wang, et al., 2008; Zhang et al., 2007; Zhang, Wang, Ye, Zhou, & Wang, 2009; Zhang, Wang, Ye, et al., 2008; Zhang, Wang, Ye, Zhou, et al., 2010). In these schemes, the downsampling and upsampling cause distortion in signal frequency spectra and deteriorate the learning performance. To solve this problem, some considerations in data processing including signal extension, anti-aliasing, and anti-imaging are investigated in this paper and integrated with these schemes. The remainder of the paper is organized as follows: Section 2 discusses the idea of multirate ILC with its necessary signal processing techniques. Section 3 enhances several multirate ILC schemes by applying the techniques in Section 2. Each of them is followed by another one having better tracking performance. Section 4 presents a series of experimental results of the proposed multirate ILC schemes and their performances are compared, which is followed by concluding remarks in Section 5.

2. Downsampled learning

The downsampled learning can deal with both single input single output (SISO) and multiple-input multiple-output (MIMO) systems. For simplicity, consider a discrete-time linear single input single output (SISO) system

$$\begin{cases} x_{f,j}(k+1) = A_f x_{f,j}(k) + B_f u_{f,j}(k) + w_{f,j}(k) \\ y_{f,j}(k) = C_f x_{f,j}(k) + v_{f,j}(k) \end{cases} \tag{2}$$

with a one-step-ahead learning law in serial configuration of ILC:

$$\begin{cases} u_{f,j}(k) = y_d(k) + u_{L,f,j}(k) \\ u_{L,f,j+1}(k) = u_{L,f,j}(k) + \Gamma e_{f,j}(k+1) \end{cases} \tag{3}$$

where $k\in[0,p-1]$, p is the number of total sampling points of a given trajectory to be followed, state $x_{f,j}$ is a n dimensional vector, input $u_{f,j}$ and output $y_{f,j}$ are both scalars, subscript j is the iteration index, f denotes the feedback system sampling rate, and $w_{f,j}$ and $v_{f,j}$ are the repeated state disturbances and output disturbances, respectively. The error is $e_{f,j}(k) = y_d(k) - y_{f,j}(k)$ with y_d as the desired trajectory. Γ is the learning gain. It is worth noting that for other ILC schemes, the downsampled learning scheme is also applicable.

With the assumption of same initial state for all trials, we have

$$e_{f,j+1} = Q e_{f,j} \tag{4}$$