Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 50-1 (2017) 4825–4830

Data-Driven Progressive and Iterative Learning Control ⋆

Cheng-Wei Chen * Tsu-Chin Tsao **

* *Department of Mechanical and Aerospace Engineering, UCLA, Los Angeles, CA (e-mail: jwster@ucla.edu).*
** *Department of Mechanical and Aerospace Engineering, UCLA, Los Angeles, CA (e-mail: ttsao@ucla.edu).*

Abstract: This paper addresses the error convergence rate of data-driven iterative learning control (ILC) for single-input-single-output (SISO) systems. Since the error convergence rate depends on the learning filter, which ideally should invert the plant dynamics, the challenge lies in creating the ILC learning filters that approximate the plant inverse without having the plant model. Zero-phase or time-reversal filtering ILC is applied to track smoothened impulse, where the learning filter is progressively updated while trajectory learning proceeds. The approach drastically accelerates the error convergence rate of the time-reversal based ILC. The progression of the ILC learning filter brings an additional degree of freedom for the learning filter design with proven stability properties. Simulation results for tracking a chirp reference on a linear motor positioning system demonstrate the effectiveness of the approach.

Keywords: Iterative learning control, dynamic inversion, tracking systems, feedforward control

1. INTRODUCTION

Iterative learning control (ILC), which iteratively updates control inputs, has been shown to achieve excellent tracking performance for applications that execute the same task multiple times [Bristow et al. (2006)]. Because the reference trajectory remains unchanged from one iteration to the next, the error information from the previous runs are used to anticipate the control signals and therefore high-performance tracking is enabled once converged. After decades of development, ILC has been applied to various industrial applications, such as Computer-Numerical-Control machining [Krishnamoorthy and Tsao (2004)], wafer stage motion systems [Barton and Alleyne (2008)], robotics [Tayebi (2004); Wallen et al. (2011)], the pulsed linear accelerators in the free electron laser [Rezaeizadeh and Schilcher (2008); Kirchhoff et al. (2008)], and chemical processing [Chin et al. (2004)].

The stability and error convergence rate of ILC algorithms rely on the selection of learning filter. Model-based approaches use numerical optimization [Gunnarsson and Norrlöf (2001)] or non-minimum phase plant inversion technologies, such as zero-phase-error tracking controller (ZPETC) [Tomizuka (1987)], zero-magnitude-error tracking controller (ZMETC) [Rigney et al. (2009)], and direct inversion [Chang and Tsao (2014)], to come up with the learning filter that meets the stability criteria and provides fast error convergence. Excluding the effect of model uncertainty, data-driven ILC is popular when an accurate dynamic model is not available. Time-reversal phase cancellation [Gustafsson (1996); Ye and Wang (2005); Bolder and Oomen (2015)] ensures the stability of iterative process by assigning the adjoint system with a sufficiently small learning gain as the learning filter. This conservative updating, however, results in slow error convergence and makes itself less practical for applications.

To improve convergence rate of data-driven ILC algorithms, a progressive approach is proposed to *learn* and *update* the learning filter. Inspired by the findings that dynamic inversion can be reconstructed by any stable ILC algorithm [Chen and Tsao (2016)], we initialize from a small learning gain and then use the *underconstructed* inversion to update the learning filter for next progression. While proceeding this progressive evolution, the dynamic inversion is gradually reconstructed and provides much faster ILC convergence rate than the origin. This approach reveals another degree of freedom for ILC algorithms. For any stable ILC, iteration of updating input signals brings smaller tracking error, while progression of learning filter improves error convergence rate.

The remainder of the paper is organized as follows: the problem definition and theoretical background are given in Section 2; the progression of ILC algorithm is proposed in Section 3; Section 4 demonstrates the simulation results on a high-order linear motor model; the concluding remarks are given in Section 5.

2. PROBLEM DEFINITION AND PRELIMINARIES

2.1 Iterative Learning Control (ILC)

For any stable single-input-single-output (SISO) discrete-time systems G, a generic ILC updating law can be expressed as

$$u_{j+1} = u_j + F_{ILC}(z)\,(r - y_j) \tag{1}$$

⋆ This work was supported in part by funds from UCLA Jules Stein Institute, Matech Industries, LTD. Taiwan, and the National Institute of Health, R21 Grant No. 5R21EY024065-02.

10.1016/j.ifacol.2017.08.969

where j indicates the iteration number, r is the tracking reference with length of N, u stands for the control input, and y is the output of the plant G. F_{ILC}, also denoted as learning filter, determines the stability of the iterative process and the error convergence rate.

To clarify, the capital letters (e.g. G, F_{ILC}) in the rest of this paper are the frequency-domain representation of the dynamics; the capital letters followed by the symbol (z) represent their filter form in z-domain; lower case letters (e.g. r, y) denote either time-domain vectors or the impulse responses of the corresponding dynamics.

Assumption 1. The finite time window length N is sufficiently large so that it covers the reference trajectory and the impulse response of the plant dynamics with sufficient number of zero padded on both ends. This assumption makes it possible to use the Z-transform of the signals and systems to represent and analyze the ILC operations in time and frequency domain.

After exploiting the iterative structure, the control input u at the j-th iteration can be rewritten as

$$u_j = [1 - (1 - F_{ILC}(z)G(z))^j]G^{-1}(z)r \tag{2}$$

Note, the G^{-1} term in Eq. 2 is algebraically canceled after multiplying the $[1-(1-F_{ILC}(z)G(z))^j]$ term. Therefore, u_j is bounded and stable even if G has non-minimum phase zeros. The corresponding tracking error at j-th iteration is

$$e_j = [1 - F_{ILC}(z)G(z)]^j r \tag{3}$$

Stability and gradient of ILC has been well-documented [Bristow et al. (2006)]. Particularly, [Norrlöf and Gunnarsson (2002)] has proven that Eq. 2 and Eq. 3 are asymptotically stable iff

$$|1 - F_{ILC}G|_\infty < 1 \tag{4}$$

Remark 1. (Error convergence rate). Eq. 4 not only specifies the stability by using the selected learning filter F_{ILC}, but also implies how fast the tracking error converges. When F_{ILC} perfectly models the dynamic inversion of G, the convergence can be achieved within a single iteration. On the other hand, if the magnitude of F_{ILC} approximates to or equals zero, the tracking error is slowly decreased or even not improved.

2.2 ILC-based Feedforward Filter (ILCFF)

An interesting findings in [Chen and Tsao (2016)] motivates the proposed progression of ILC. It is reported that dynamic inversion can be reconstructed from any stable ILC algorithm, either model-based or data-based, by learning a filtered delta function. After the control input u converges and the error between the output y and the targeted impulse m is less than the error tolerance ϵ, the learned input signal u is used to represent the impulse response of the dynamic inversion F, such that

$$\begin{aligned} GF = GU = Y \\ = M + E \cong M \end{aligned} \tag{5}$$

Note, M is the frequency domain representation of the filtered impulse m and forms a zero-phase low-pass filter

$$M(z) = H(z)H(z^{-1}) \tag{6}$$

where $H(z)$ could be an arbitrary low-pass filter.

Assumption 2. The zero-phase low-pass filter $M(z)$ does not have zeros on unity circle. Therefore $M^{-1}(z)$ exists.

The analysis and performance of the reconstructed inverse filter have already been studied in [Chen and Tsao (2016)] therefore they are ignored here to save space. Most importantly, the reconstructed dynamic inversion can be directly used as a feedforward filter, which is preferable when the tracking reference r is changing and therefore ILC can not be applied.

Inspired by the excellent feedforward tracking performance done by the reconstructed inverse filter, we were thinking the possibility to first use a data-based ILC algorithm to reconstruct a dynamic inversion, and then plug the reconstructed inversion back into the generic ILC as the learning filter to improve the performance of the initial ILC algorithm. As a result, a decent data-based ILC algorithm can be formulated without any *a priori* knowledge regarding the plant dynamics.

2.3 Time-Reversal Phase cancellation

To enable the reconstruction of inverse filter, [Chen and Tsao (2016)] used the model-based ZPETC [Tomizuka (1987)] as the learning filter in the generic ILC. Applying the adjoint system, ZPETC was used to stabilize repetitive controller [Tsao and Chew (1989)] and was shown could be used as the learning filter for ILC algorithm [Longman (2000)]. The convergence and robustness properties of adjoint operator were studied in the norm optimal ILC [Kinosita et al. (2002); Ratcliffe et al. (2008); Owens et al. (2009)], where a precise plant model is applied.

For model-free ILC design, time-reversal technique was proposed to realize the adjoint operator [Gustafsson (1996); Ye and Wang (2005)]. This approach was later extended to handle point-to-point tracking [Potsaid and Wen (2004)], non-minimum phase plants [Freeman et al. (2007)], stochastic disturbance [Butcher et al. (2008)], and multivariable systems [Bolder and Oomen (2015)].

Definition 2. (Time-reversal operator). A time reversal operator $\mathcal{R}$ which flips a sequence of signal is defined as an N-by-N involutory permutation matrix such that

$$y^*(i) = y(N - i) = \mathcal{R}y(i) \tag{7}$$

Definition 3. (Adjoint system). G^*, with an input y and an output w, is the adjoint system of the SISO plant G where

$$w(i) = G^*(z)y(i) = \mathcal{R}G(z)\mathcal{R}y(i) \tag{8}$$

Because G^*G forms a system with positive magnitudes and zero phase angles, G^* can stabilize the model-free iterative process after scaled by a constant learning gain α:

$$F_{ILC}(z) = \alpha G^*(z) \tag{9}$$

When α is sufficiently small, the stability criteria of Eq. 4 is met and therefore the stability of the time-reversal based ILC is guaranteed. However, small α results in conservative updating and thus slows down the error convergence rate. To improve, a data-driven progressive method is proposed to update the conservative learning filter and dynamic inversion, such that a faster convergence rate can be evolved.

3. PROGRESSION OF ILC

The formulation of the proposed progressive algorithm is presented in this section. To clarify, the index of the