Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

On Improving Transient Behavior and Steady-State Performance of Model-free Iterative Learning Control ⋆

Geng-Hao Zhang * **Cheng-Wei Chen** **

* *Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan (e-mail: r07921105@ntu.edu.tw).*
** *Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan (e-mail: cwchenee@ntu.edu.tw).*

Abstract: A novel model-free iterative learning control algorithm is proposed in this paper to improve both the robustness against output disturbances and the tracking performance in steady-state. For model-free ILC, several methods have been investigated, such as the time-reversal error filtering, the Model-Free Inversion-based Iterative Control (MFIIC), and the Non-Linear Inversion-based Iterative Control (NLIIC). However, the time-reversal error filtering has a conservative learning rate. Other two methods, although with much faster error convergence, have either a high noise sensitivity or a non-optimized steady-state. To improve the performance and robustness of model-free ILC, we apply the time-reversal based ILC and recursively accelerate its error convergence using the online identified learning filter. The effectiveness of the proposed algorithm has been validated by a numerical simulation. The proposed approach not only improves the transient response of the MFIIC, but achieves lower tracking error in steady-state compared to that of the NLIIC.

Keywords: Iterative learning control; Model-free; Convergence analysis.

1. INTRODUCTION

Iterative learning control (ILC), which operates the same task repeatedly and updates the control input according to the previous trial data, has been applied to tracking control applications for many years [Ahn et al. (2007)]. Since the learning operation can be non-casual, ILC has been shown to achieve better tracking performance compared to feedback control approaches [Bristow et al. (2006)].

To reduce the tracking error iteratively, the model-based ILC [Lee et al. (1994)] uses the system inverse model as the learning filter to conduct the learning process. For model-based approaches, if the system model is identified accurately, the learning process would converge within few iterations [Teng and Tsao (2015)]. However, inversion-based methods require delicate modeling process, which is not practical for industrial applications. Thus, the development of model-free ILC becomes more and more popular in the ILC research community [Janssens et al. (2011)].

For the model-free approaches, the PD-type ILC [Arimoto et al. (1984); Chen and Hwang (2006)] is the simplest way to implement. However, monotonic convergence condition is not always satisfied by tuning the PD-parameters [Moore et al. (2005)], and the tuning process may be time-consuming and damage the system. Time-reversal based ILC [Ye and Wang (2005)] uses the adjoint operator of the system as the learning filter. The control updating law is implemented by filtering the reversed error signals by the system itself. However, the learning rate is usually slow [Chen et al. (2020)].

To achieve a fast error convergence rate, the model-free inversion-based iterative learning control (MFIIC) [Kim and Zou (2012)] applies the point-by-point division over the frequency response function of the system input and the output signals to estimate the inverse dynamics. Nevertheless, the output disturbances present in the denominator of the computation. Once unpredicted output disturbances are introduced, the magnitude of the denominator may be greatly influenced in the high-frequency range. This unpredictable phenomenon leads to a poor transient learning behavior [De Rozario and Oomen (2018)]. To improve the learning transient performance, NLIIC [de Rozario and Oomen (2019)] uses an adaptive learning gain to avoid the division by small numbers. However, since the adaptive learning gain shuts down the updating for the high-frequency components, the tracking performance in steady-state is degraded.

To overcome the difficulties mentioned above, a model-free learning algorithm is proposed. The proposed algorithm is based on the time-reversal based ILC, and we apply a data-based learning filter to update the control input every n iterations. This approach not only accelerates the convergence rate of time-reversal based ILC. Compared to the MFIIC and NLIIC, our method provides a more flexible updating law that gains the robustness against output disturbances and the tracking performance in steady-state.

⋆ This work was financially supported from the Young Scholar Fellowship Program by Ministry of Science and Technology (MOST) in Taiwan, under Grant MOST 108-2636-E-002-007.

Copyright lies with the authors

The remainder of this paper is organized as follows: Section 2 illustrates the basic iterative learning control algorithm and the existing model-free ILC approaches. Section 3 provides a derivation and analysis of the proposed model-free learning algorithm. Section 4 presents the simulation results to validate the effectiveness of the proposed algorithm. Finally, Section 5 summarizes the main results and contributions of this paper.

2. PROBLEM FORMULATION

2.1 Iterative Learning Control

For a SISO, LTI system $G(z)$, which is represented in z-domain and assumed to be asymptotically stable, the ILC learning algorithm can be written in the following form:

$$\begin{aligned} u_{j+1}(k) &= u_j(k) + L(z)[r(k) - y_j(k)] \\ &= u_j(k) + L(z)e_j(k) \end{aligned} \tag{1}$$

where k is the time index of the discrete-time signal, the sub-index j represents the jth iteration of learning process, r is the trial-invariant reference to be tracked with time interval N, u represents the control input of $G(z)$, y is the output of $G(z)$, e is the tracking error to be reduced, and $L(z)$ is the learning filter to guarantee the stable learning.

Assumption 1. Since the z-transform and Fourier transformation of a time sequence is computed over an infinite time interval, all the time signals in this paper are assumed to have infinite length (i.e. $N \to \infty$) to meet the requirement of frequency-domain analysis .

The ILC stability condition in frequency domain is:

$$\|I - G(z)L(z)\|_\infty < 1 \tag{2}$$

where $\|\cdot\|_\infty$ denotes $\mathcal{H}_\infty$-norm. It had been proven when the learning filter $L(z)$ satisfies Eq.2, the control input u_∞ and tracking error e_∞ will converge and thus the learning is stable [Norrlöf and Gunnarsson (2002)] .

Remark 1. To meet the infinite long time signal assumption, the zero-padding technique is applied to extend the reference signal with sufficient long zeros on the both ends to avoid leakage in frequency-domain computation.

2.2 Model-free ILC

To satisfy the stability criteria, there are several ways to implement the model-free leaning law. Here the time-reversal based ILC and MFIIC approaches are introduced.

Time-reversal based ILC

Consider the adjoint-based ILC [Ye and Wang (2005); Owens et al. (2009)], the learning filter $L(z) = \alpha G^*(z)$ is chosen with a sufficiently small learning gain $\alpha > 0$ to guarantee the stability condition.

To realize the adjoint-based learning law:

$$u_{j+1}(k) = u_j(k) + \alpha G^*(z)e_j(k) \tag{3}$$

the term $\alpha G^*(z)e_j(k)$ can be obtained by using the reversed time filtering technique as shown in [Ye and Wang (2005)]:

(1) Reverse the error signal: $e_{j1}(k) = e_j(N - k)$;
(2) Feed the reversed error signal to the system: $e_{j2}(k) = G(z)e_{j1}(k)$;
(3) Reverse $e_{j2}(k)$ again, and multiply with α: $e_{j3}(k) = \alpha e_{j2}(N - k)$
(4) Conduct the learning law: $u_{j+1}(k) = u_j(k) + e_{j3}(k)$;

For the adjoint-based learning approach, the robustness can be improved compared with the inversion-based approach, especially considering the effect of high frequency model uncertainty [Owens et al. (2009)]. However, since the convergence rate is limited by the small learning gain, this disadvantage leads to a critical issue of the time-consuming learning process in real application.

Model-free Inversion-based Iterative Control

Consider the SISO, LTI and stable system $y_j(k) = G(z)u_j(k)$, the model-free inversion-based learning law can be written as:

$$U_{j+1}(e^{j\omega}) = \begin{cases} U_j(e^{j\omega}) + \rho_j \dfrac{U_j(e^{j\omega})}{Y_j(e^{j\omega})} E_j(e^{j\omega}), \\ \quad \text{if } Y_j(e^{j\omega}) \neq 0 \text{ and } R(e^{j\omega}) \neq 0; \\ U_j(e^{j\omega}); \text{ otherwise} \end{cases} \tag{4}$$

where $U_j(e^{j\omega})$ and $Y_j(e^{j\omega})$ is the frequency-domain representation of $u_j(k)$ and $y_j(k)$ respectively; and the learning gain ρ_j is equal to 1 for MFIIC [Kim and Zou (2012)] and $\rho_j = f(|Y_j(e^{j\omega})|)$ is a function of $|Y_j(e^{j\omega})|$ for NLIIC [de Rozario and Oomen (2019)] to improve the robustness.

For this approach, since the estimated learning filter is inversion-based, the learning will converge after few iterations. However, due to the learning filter is constructed by measured data, while the unpredicted disturbances dominate the output signal, the updated control input may be dramatically amplified in some frequency components [De Rozario and Oomen (2018)]. Moreover, since the noisy learning filter is conducted iteration by iteration over the whole learning process, while the adaptive learning gain is introduced [de Rozario and Oomen (2019)], the tracking performance in steady-state is limited.

In this paper. we propose a novel method to remedy the difficulties for the model-free ILC mentioned above. Specifically, the proposed method is expected to

(1) Improve the learning transient against output disturbances;
(2) Achieve lower tracking error in steady-state.

3. MODEL-FREE ILC WITH RECURSIVE CONVERGENCE ACCELERATION

The main idea of the proposed algorithm is to formulate the ILC recursive equation (Eq. 1) into a closed-form, namely an equation describes recursive convergence rate acceleration. By exploiting such the equation, the learning rate of any stable learning law can be further accelerated.

The derivation of the proposed algorithm will be discussed in section 3.1, and the stability and robustness of the proposed algorithm will be analysed in section 3.2 and 3.3.

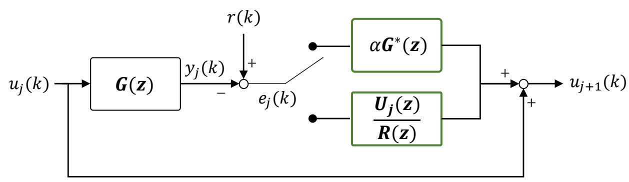


Fig. 1. The control architecture of the proposed algorithm. The learning is switched to the acceleration filter $\frac{U_j(z)}{R(z)}$ for every $n \in \mathbb{N}$ iterations; otherwise, the control input is updated by the time-reversal based ILC.

3.1 Algorithm Formulation

Since the ILC learning law is a recurrence relation, given the initial term $u_0 = L(z)r(k)$, Eq. 1 can be reformulated as the following form:

$$u_{j-1}(k) = \frac{1 - [1 - L(z)G(z)]^j}{G(z)} r(k) \tag{5}$$

$$e_{j-1}(k) = [1 - G(z)L(z)]^j r(k) \tag{6}$$

Assume all the discrete time signals are with infinity length, the time-domain signal $x(k)$ can be transformed into the z-domain representation:

$$X(z) := \sum_{k=-\infty}^{\infty} x(k) z^{-k} \tag{7}$$

As a result, Eq. 5 and Eq. 6 can be rewritten as:

$$U_{j-1}(z) = \frac{1 - [1 - L(z)G(z)]^j}{G(z)} R(z) \tag{8}$$

$$E_{j-1}(z) = [1 - G(z)L(z)]^j R(z) \tag{9}$$

where $U_{j-1}(z)$ denotes the z-domain representation of $u_{j-1}(k)$, and $R(z)$ and $E_{j-1}(z)$ are as so on. Inspired by the recursive equation, if the sub-index j is replaced with $2j$, then Eq. 8 becomes

$$U_{2j-1}(z) = \frac{1 - [1 - L(z)G(z)]^{2j}}{G(z)} R(z) \tag{10}$$

Combine Eq. 9 and Eq. 10,

$$U_{2j-1}(z) = \frac{1 - [\frac{E_{j-1}(z)}{R(z)}]^2}{G(z)} R(z) \tag{11}$$

$$= 2U_{j-1}(z) - \frac{U_{j-1}(z)}{R(z)} Y_{j-1}(z) \tag{12}$$

Reformulate Eq. 12 into the ILC updated form, we can obtain

$$U_{2j-1}(z) = U_{j-1}(z) + \frac{U_{j-1}(z)}{R(z)} E_{j-1}(z) \tag{13}$$

Eq. 13 implies if the learning filter $\frac{U_{j-1}(z)}{R(z)}$ is assigned, the learning rate will be dramatically improved.

To implement this observation, consider the SISO, LTI and pre-stabilized system $G(z)$, and the initial learning filter $L_0(z)$ is chosen to satisfy the stability criteria (Eq. 2). The proposed control input updating approach utilizes Eq. 13 to update the control input every $n \in \mathbb{N}$ iterations, and the rest iterations update the control input via $L_0(z)$. The control architecture is as shown in Fig. 1, and the procedure of the proposed algorithm is summarized below:

Algorithm 1 Model-free ILC with Recursive Convergence Acceleration

1: Initialize $j = 0$, $u_0(k) = r(k)$, $L_0(z) = \alpha G^*(z)$.
2: Feed $u_j(k)$ to $G(z)$ to obtain the output data $y_j(k)$; Record the tracking error $e_j(k) = r(k) - y_j(k)$; If tracking error is small enough within the tolerance, then stop the learning; otherwise, go to the next step.
3: If $j = 0$ or $(j \ \boldsymbol{mod}\ n) \neq 0$, go to step 4; otherwise, go to step 5;
4: Conduct the time-reversal based ILC: $u_{j+1}(k) = u_j(k) + \alpha G^*(z) e_j(k)$; Set $j \leftarrow j + 1$, go to step 2;
5: Apply DFT to obtain the frequency-domain signal $R(e^{j\omega})$, $U_j(e^{j\omega})$ and $E_j(e^{j\omega})$;
6: Conduct the proposed updating law (Eq. 13) over the discrete-frequency interval:

$$U_{j+1}(e^{j\omega}) = \begin{cases} U_j(e^{j\omega}) + \dfrac{U_j(e^{j\omega})}{R(e^{j\omega})} E_j(e^{j\omega}), \\ \text{if } |R(e^{j\omega})| > threshold; \\ U_j(e^{j\omega}); \text{ otherwise} \end{cases} \tag{14}$$

Do inverse DFT to obtain the time-domain sequence $u_{j+1}(k)$; Set $j \leftarrow j + 1$, go to step 2;

Note that the learning threshold of Eq. 14 must be set if the output disturbances are considered. How to set the learning threshold to maintain the stable learning will be discussed and analysed in Section 3.3.

3.2 Stability Analysis

To prove the proposed ILC algorithm is stable, some assumptions and definitions are stated below:

Definition 1. (Error Convergence Rate) Consider the stable system $G(z)$ and the learning filter $L(z)$, the error convergence rate of ILC is defined as:

$$\gamma := \|I - G(z)L(z)\|_\infty \tag{15}$$

The smaller the value of γ is, the faster the tracking error converges.

Definition 2. (Symbol Convention) The learning filter updated at each pn iteration is denoted as $L_{pn}(z)$, where $p \in \mathbb{N}$ is the number of updating times.

Assumption 2. The initial learning filter $L_0(z)$ is stable, i.e., $\gamma_0 := \|I - G(z)L_0(z)\|_\infty < 1$

Theorem 1. (Stability of Algorithm 1.)

Assume the learning filter conducted at each pn iteration by algorithm 1 is:

$$L_{pn}(z) = \frac{U_{pn}(z)}{R(z)} \tag{16}$$

where $U_{pn}(z)$ is the control input applying algorithm 1 at pn iteration in z-domain representation, $p, n \in \mathbb{N}$, then:

$$\gamma_{pn} := \|I - G(z)L_{pn}(z)\|_{\infty} \le (\gamma_0)^{2^{p-1}(n+1)} < 1 \quad (17)$$

Proof:

According to Eq. 8 and Eq. 13, the control input conducted by algorithm 1 at each pn iteration can be written as:

$$U_{pn}(z) = \frac{1 - [1 - L_0(z)G(z)]^{(2^p-1)n+2^{p-1}}}{G(z)} R(z) \quad (18)$$

Hence the learning filter $L_{pn}(z)$ becomes:

$$L_{pn}(z) = \frac{U_{pn}(z)}{R(z)} = \frac{1 - [1 - L(z)G(z)]^{(2^p-1)n+2^{p-1}}}{G(z)} \quad (19)$$

Combine Eq. 2 and Eq. 19, the stability criteria becomes:

$$\gamma_{pn} := \|I - G(z)L_{pn}(z)\|_{\infty} \quad (20)$$

$$\le \|I - G(z)L_0(z)\|_{\infty}^{(2^p-1)n+2^{p-1}} \quad (21)$$

Since $2^p - 1 \ge 2^{p-1}$ for $p \in \mathbb{N}$, therefore

$$(2^p - 1)n + 2^{p-1} \ge 2^{p-1}n + 2^{p-1} = 2^{p-1}(n+1) \quad (22)$$

From Eq. 21 and Eq. 22,

$$\gamma_{pn} \le (\gamma_0)^{(2^p-1)n+2^{p-1}} \le (\gamma_0)^{2^{p-1}(n+1)} < 1 \quad (23)$$

■

Theorem 1 implies if the pre-selected learning filter $L_0(z)$ is stable, then the acceleration learning filter $L_{pn}(z)$ not only satisfies the stability criteria, but also accelerates the learning rate exponentially at each pn iteration, $p, n \in \mathbb{N}$.

3.3 Robustness Analysis

To guarantee the tracking error conducted by the proposed learning algorithm converges monotonically when the output disturbances are considered, assume the system dynamics of a stable $G(z)$ becomes:

$$y_j(k) = G(z)u_j(k) + d(k) \quad (24)$$

where $d(k)$ is the output disturbance.

Assumption 3. The output disturbance $d(k)$ is trial-invariant, and can be represented in z-domain and frequency-domain as $D(z)$ and $D(e^{j\omega})$ respectively. $|D(e^{j\omega})|$ is bounded by δ, i.e., $\delta := max(|D(e^{j\omega})|)$ $\forall \omega \in (-\pi, \pi)$.

Theorem 2. (Learning threshold of Algorithm 1.) Consider the system dynamics of SISO, LTI system $G(z)$ in frequency-domain: $Y_j(e^{j\omega}) = G(e^{j\omega})U_j(e^{j\omega}) + D(e^{j\omega})$, where $|D(e^{j\omega})|$ is bounded by δ, i.e., $\delta := max(|D(e^{j\omega})|)$ $\forall \omega \in (-\pi, \pi)$. The necessary condition for the monotonic convergence of the learning law Eq. 14 is

$$|R(e^{j\omega})| > |E_j(e^{j\omega})| + \delta \quad (25)$$

where $R(e^{j\omega})$ is the reference trajectory and $E_j(e^{j\omega}) := R(e^{j\omega}) - Y_j(e^{j\omega})$ is the tracking error.

Proof:

Since the tracking error at jth iteration is:

$$E_j(e^{j\omega}) = R(e^{j\omega}) - G(e^{j\omega})U_j(e^{j\omega}) - D(e^{j\omega}) \quad (26)$$

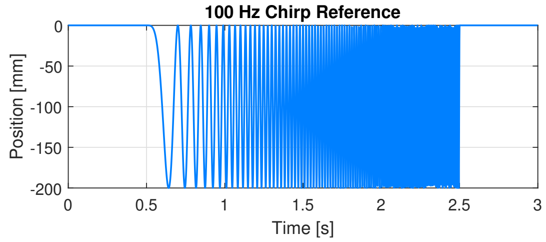


Fig. 2. The chirp reference with the amplitude of 100 mm and a maximum frequency component of 100 Hz.

the tracking error at $(j+1)$th iteration conducted by the learning law Eq. 14 becomes:

$$E_{j+1}(e^{j\omega}) = R(e^{j\omega}) - G(e^{j\omega})U_{j+1}(e^{j\omega}) - D(e^{j\omega}) \quad (27)$$

$$= [\frac{E_j(e^{j\omega}) + D(e^{j\omega})}{R(e^{j\omega})}]E_j(e^{j\omega}) \quad (28)$$

Define

$$\kappa_j := \frac{E_j(e^{j\omega}) + D(e^{j\omega})}{R(e^{j\omega})} \quad (29)$$

then

$$E_{j+1}(e^{j\omega}) = \kappa_j E_j(e^{j\omega}) \quad (30)$$

Therefore,

$$|E_{j+1}(e^{j\omega})| \le |\kappa_j||E_j(e^{j\omega})| \quad (31)$$

To guarantee the tracking error is monotonically decaying, the following condition must be satisfied:

$$|\kappa_j| := |\frac{E_j(e^{j\omega}) + D(e^{j\omega})}{R(e^{j\omega})}| < 1 \quad (32)$$

Hence the necessary condition for the monotonic convergence of learning law Eq. 14 is:

$$|R(e^{j\omega})| > |E_j(e^{j\omega}) + D(e^{j\omega})| \quad (33)$$

Since

$$|E_j(e^{j\omega})| + \delta \ge |E_j(e^{j\omega})| + |D(e^{j\omega})| \ge |E_j(e^{j\omega}) + D(e^{j\omega})| \quad (34)$$

If the NSR condition Eq. 25 is hold, the monotonic convergence of updating law Eq. 14 is guaranteed.

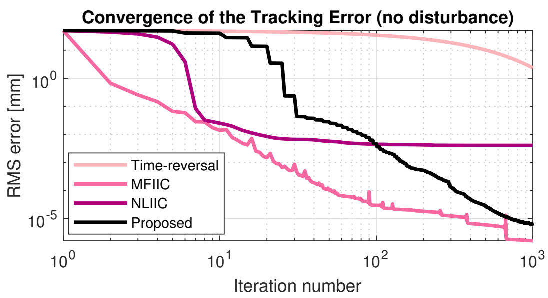
■

From *Theorem* 2, the learning threshold to maintain the stable learning at each pn iteration can be derived. *Theorem* 2 implies if the learning threshold (Eq. 25) is set, the tracking will be shut down when the frequency components do not reach the threshold. However, since the rest iterations use time-reversal based ILC to reduce the tracking error, these frequency components can still be learned. Such a dual-mode learning not only provides a more robust learning algorithm than MFIIC, but also achieves better steady-state tracking performance compared to [de Rozario and Oomen (2019)].

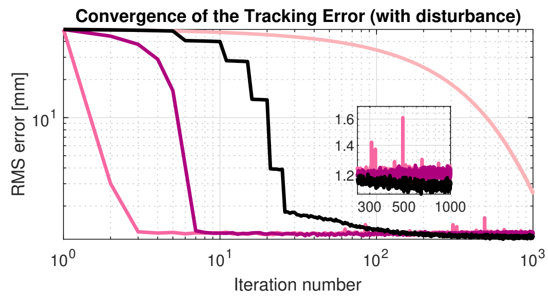
4. SIMULATION RESULTS

4.1 Simulation Set-up

To verify the effectiveness of the proposed algorithm, a numerical example of the system $G(z)$ is performed.



(a)



(b)

Fig. 3. The convergence of the ILC tracking error. (a) The case with no output disturbance. (b)The case with the output disturbance $d(k) \in N(0, 1)$.

As mentioned previously, $G(z)$ is pre-stabilized, and the system applied in the simulation is

$$G(z) = \frac{0.2(z - 0.9)(z - 0.995)}{(z + 0.1803)(z^2 - 1.982z + 0.9824)} \tag{35}$$

The chirp signal (linear swept-frequency cosine) is chosen as the reference trajectory in the simulation. Here the chirp trajectory with the amplitude of 100 mm and a maximum frequency component of 100 Hz is applied as the illustrated example in this paper, as shown in Fig. 2. Note that to reduce the influence of the frequency leakage as pointed out in Sec. 2, the chirp reference is padded with sufficient long zeros on the both ends.

To compare the performance of each model-free ILC, there are 4 cases are performed, each case is performed the updating law for 1000 iterations, and all the initial control inputs are set as $u_0(k) = r(k)$. The first case is to conduct the time-reversal based ILC, where $L_0(z) = \alpha G^*(z)$, $\alpha = 0.1$ is chosen; the second case is to conduct the MFIIC; the third case is to apply the NLIIC [de Rozario and Oomen (2019)] to enhance the robustness of MFIIC; and the fourth case is to perform the proposed algorithm, where the initial learning filter $L_0(z)$ is same as the case 1, and the updating period $n = 5$, i.e., the acceleration updating occurs at 5th, 10th,..., 1000th iterations.

4.2 Ideal Case Without Output Disturbance

Consider the ideal case, where the output disturbance is neglected in this section. The only factor impacts the unideal learning for the disturbance-free case in the simulation is the frequency-leakage problem as pointed out in section 2.

Fig. 3(a) shows the error convergence of each learning law with no output disturbance. From the simulation result, it can be observed the convergence rate of time-reversal based ILC is the slowest. The convergence rate of the proposed learning law is accelerated every 5 iterations compared with time-reversal based approach, hence the learning rate is further improved. Although the convergence rate of MFIIC is faster than time-reversal based ILC, however, the unpredictable learning transient may appear in some iterations. NLIIC solve the robustness issue of MFIIC, however, the tracking is limited by the adaptive learning gain, hence the tracking performance in steady-state is much worse than MFIIC and proposed algorithm.

On the other hand, although the convergence rate of the proposed learning law is slower than MFIIC and NLIIC, however, the learning transient is improved by avoiding the noisy signal division. Moreover, the proposed algorithm provide better tracking performance in steady-state compared to NLIIC. From Fig. 5(a), it can be observed the NLIIC is suffered from the influence of

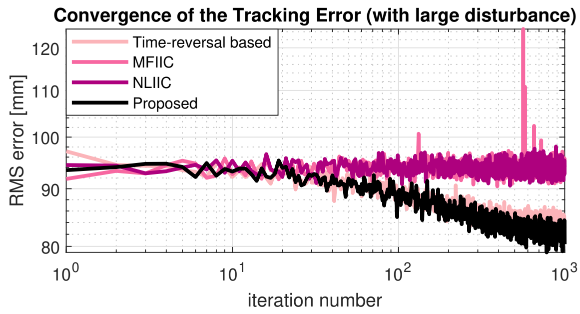
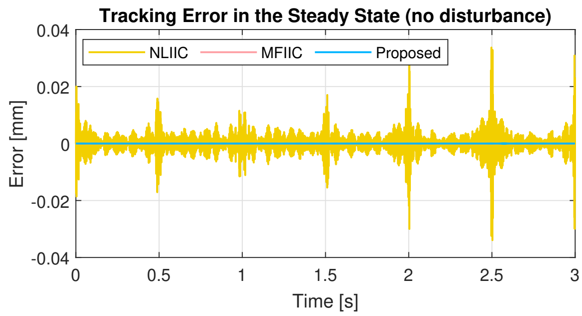
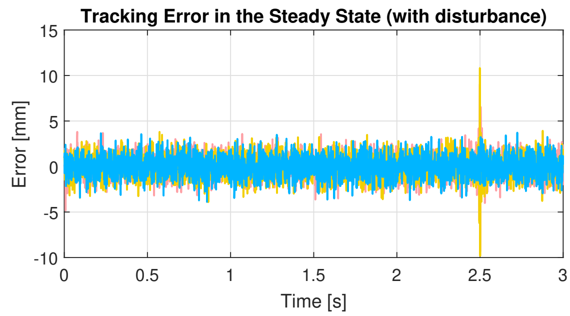


Fig. 4. The convergence of the ILC tracking error when the output disturbance $d(k) \in N(0, 80)$. Both MFIIC and NLIIC do not converge.



(a)



(b)

Fig. 5. The ILC tracking error in steady-state. (a) The case with no output disturbance. (b) The case with the output disturbance $d(k) \in N(0, 1)$.

Table 1. The tracking error obtained from different methods

Method	No Disturbance		With Disturbance	
	RMSE [mm]	MaxE [mm]	RMSE [mm]	MaxE [mm]
Time-reversal	2.33	6.13	2.54	8.15
MFIIC	1.64e-06	1.75e-06	1.24	9.61
NLIIC	0.00410	0.0339	1.22	13.8
Proposed	6.25e-06	1.64e-05	1.12	4.52

frequency leakage problem. The dual-mode learning of the proposed algorithm further reduces the tracking error caused by the frequency leakage, hence the steady-state performance is improved. The RMS error and maximum error of each learning law at the 1000th iteration with no output disturbance is shown in Table 1.

4.3 Output Tracking with Trial-variant Disturbances

Consider the trial-variant normally distributed random time sequences (the MATLAB function: randn) are applied as the output disturbances, the tracking result of each model-free learning law is shown in this section.

Fig. 3(b) shows the error convergence of each learning law with the output disturbances, where the standard deviations of the disturbances are 1 mm and the mean values are 0 mm (i.e., $d(k) \in N(0, 1)$). From the simulation results, it can be seen the convergence rate of the proposed algorithm is dramatically improved compared with time-reversal based ILC, and the learning transient is also reduced, hence the robustness is improved. Moreover, the tracking performance at the 1000th iteration of the proposed algorithm is supreme, as shown in Table 1.

To show the robustness and the steady-state performance of the proposed algorithm outperforms that of the MFIIC and NLIIC, consider the extreme situation that the output disturbances with the standard deviations of 80 mm and the mean values of 0 mm (i.e., $d(k) \in N(0, 80)$) are applied to track the 100 mm chirp reference, as shown in Fig. 4. Although NLIIC provides a smoother learning curve than MFIIC approach, however, the tracking error doesn't converge to a lower level. On the other hand, since the proposed algorithm is based on time-reversal based ILC, the robustness and the limited steady-state tracking performance problems can be remedied.

From Fig. 5(b), it can be observed the maximum tracking error of MFIIC and NLIIC occur at around 2.5 sec, i.e., the high-frequency components of the chirp reference. The proposed algorithm not only further reduces the tracking error in high-frequency, but also provides a more robust and flexible updating law compared with MFIIC and NLIIC, hence the tracking performance is improved if the output disturbances are considered.

5. CONCLUSION

In this paper, a novel model-free iterative learning control algorithm is proposed. Applying the recursive relation of the ILC control updating law, the proposed method accelerates the learning rate of the time-reversal based ILC every $n \in \mathbb{N}$ iterations. The stability and robustness of the proposed algorithm is analyzed and discussed. According to the numerical simulation, the robustness and tracking performance of the proposed method outperforms that of the other model-free inversion-based ILC. The implementation of the proposed algorithm is left as the future work.

REFERENCES

Ahn, H.S., Chen, Y., and Moore, K.L. (2007). Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(6), 1099–1121.

Arimoto, S., Kawamura, S., and Miyazaki, F. (1984). Bettering operation of robots by learning. *Journal of Robotic systems*, 1(2), 123–140.

Bristow, D.A., Tharayil, M., and Alleyne, A.G. (2006). A survey of iterative learning control. *IEEE control systems magazine*, 26(3), 96–114.

Chen, C., Rai, S., and Tsao, T. (2020). Iterative learning of dynamic inverse filters for feedforward tracking control. *IEEE/ASME Transactions on Mechatronics*, 25(1), 349–359.

Chen, C.K. and Hwang, J. (2006). Pd-type iterative learning control for the trajectory tracking of a pneumatic xy table with disturbances. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 49(2), 520–526.

De Rozario, R. and Oomen, T. (2018). Improving transient learning behavior in model-free inversion-based iterative control with application to a desktop printer. In *2018 IEEE 15th International Workshop on Advanced Motion Control (AMC)*, 455–460.

de Rozario, R. and Oomen, T. (2019). Data-driven iterative inversion-based control: Achieving robustness through nonlinear learning. *Automatica*, 107, 342–352.

Janssens, P., Pipeleers, G., and Swevers, J. (2011). Model-free iterative learning control for lti systems and experimental validation on a linear motor test setup. In *Proceedings of the 2011 American Control Conference*, 4287–4292.

Kim, K.S. and Zou, Q. (2012). A modeling-free inversion-based iterative feedforward control for precision output tracking of linear time-invariant systems. *IEEE/ASME Transactions on Mechatronics*, 18(6), 1767–1777.

Lee, K., Bang, S., and Chang, K. (1994). Feedback-assisted iterative learning control based on an inverse process model. *Journal of Process Control*, 4(2), 77–89.

Moore, K.L., Chen, Y., and Bahl, V. (2005). Monotonically convergent iterative learning control for linear discrete-time systems. *Automatica*, 41(9), 1529–1537.

Norrlöf, M. and Gunnarsson, S. (2002). Time and frequency domain convergence properties in iterative learning control. *International Journal of Control*, 75(14), 1114–1126.

Owens, D., Hatonen, J., and Daley, S. (2009). Robust monotone gradient-based discrete-time iterative learning control. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 19(6), 634–661.

Teng, K.T. and Tsao, T.C. (2015). A comparison of inversion based iterative learning control algorithms. In *2015 American Control Conference (ACC)*, 3564–3569.

Ye, Y. and Wang, D. (2005). Zero phase learning control using reversed time input runs. *Journal of dynamic systems, measurement, and control*, 127(1), 133–139.