Feedforward Tuning by Fitting Iterative Learning Control Signal for Precision Motion Systems

Luyao Dai, Xin Li, Yu Zhu, *Member, IEEE*, and Ming Zhang

***Abstract*—In this paper, a feedforward tuning approach by fitting iterative learning control (ILC) signal for precision motion systems is proposed. The idea is that, since ILC can achieve excellent tracking performance, ILC signal contains useful information about the plant dynamics and thus can be utilized for effective tuning of model-based feedforward controller. By this method, the high performance of ILC can be attained while the sensitivity of ILC to reference trajectory variation can be avoided. In the proposed algorithm, acceleration, jerk and snap feedforward are tuned by fitting ILC signal through least squares first; then, the residual tracking error is compensated by an extra model-based feedforward controller, the structure of the controller is determined by Monte Carlo search and the parameters of the controller are also tuned by fitting ILC signal through least squares. Simulation and experiment on a precision motion system both validate the theoretical analysis and the proposed algorithm.**

***Index Terms*—precision motion systems, feedforward tuning, iterative learning control**

I. Introduction

FEEDFORWARD control is an indispensable technique for settling performance of precision motion systems [1]–[13], which determines system throughput. The most classical feedforward method is model-based feedforward such as the acceleration, jerk and snap feedforward [2]–[4], [8], zero-phase-error tracking control (ZPETC), zero-magnitude-error tracking control (ZMETC), non-minimum-phase-zero-ignore tracking control (NMPZITC) [14]–[16]. The essence of model-based feedforward control is to approximate the inversion of the plant model with a feedforward controller. The accuracy of feedforward parameters is of significant importance to achieve satisfactory tracking performance, especially for precision motion systems requiring nano-level motion accuracy like the motion stage of lithography machine [4], [5], [7], [8]. Since the true model parameters are hard to attain, data-based techniques are extensively utilized to tune model-based feedforward controller [1], [7]–[10], [13], [17].

Among various kinds of feedforward techniques, iterative learning control (ILC) is the most powerful one from the perspective of tracking performance [11], [18]–[27]. The main drawback of ILC is the sensitivity to reference trajectory variation, i.e., once the reference trajectory is changed, ILC signal has to be relearned. Therefore, model-based feedforward is still of great practical value.

The idea of this paper is that, since ILC can achieve very high tracking performance, ILC signal must contain useful information about the plant model. Instead of tuning the feedforward controller by minimizing a certain norm of the tracking error as it did in [1], [9], [10], [13], [17], in this paper, tuning of feedforward is realized by fitting ILC signal. If the ILC signal can achieve perfect tracking and can be perfectly fitted by the model-based feedforward controller chosen, the model-based feedforward controller can also achieve perfect tracking. Furthermore, since the feedforward controller is model-based, it is insensitive to reference trajectory variation.

The idea of fitting ILC signal with a model-based feedforward controller is not new, see [28], [29]. The main differences between this paper and previous works are given in section VII-E. The main contributions of this paper can be stated as follows:

1) The plant dynamics of precision motion systems is quantitatively analyzed. Based on the analysis, a model-based feedforward controller structure to achieve high tracking performance is proposed, where acceleration, jerk (the third derivative of reference trajectory), snap (the fourth derivative of reference trajectory) feedforward and an extra compensation part are utilized. The structure of the extra compensation part, which can be either an FIR or IIR filter, is determined by Monte Carlo search.

2) A feedforward tuning algorithm for the model-based feedforward controller is proposed. Acceleration, jerk and snap feedforward are first tuned by fitting ILC signal through least squares. Then the extra compensation part is searched and tuned by fitting ILC signal through least squares.

3) The extra compensation part is non-causal, the non-causality is critical for fitting and tracking performance and is quantitatively analyzed.

This paper is organized as follows: plant dynamics and control architecture are introduced in section II; the ILC algorithm utilized in this paper is introduced in section III; then simulation parameters are given in section IV; in section V, the feedforward tuning algorithm is presented with simulation examples; section VI gives experiment validation; further discussions on corresponding topics are given in section VII and section VIII gives conclusions.

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFF01011500-04 (*Corresponding author: Xin, Li*).

The authors are all with the State Key Lab of Tribology, Department of Mechanical Engineering, Tsinghua University, Beijing 100084, China, and the Beijing Key Lab of Precision/Ultra-Precision Manufacture Equipment and Control, Tsinghua University, Beijing 100084, China. (e-mail: daily16@mails.tsinghua.edu.cn; lixin_09@mail.tsinghua.edu.cn)

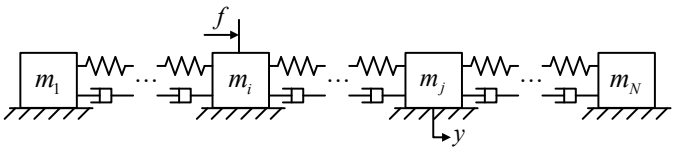


Fig. 1. Multi-mass-block model.

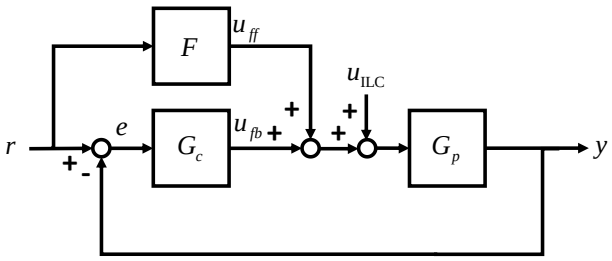


Fig. 2. Two degree-of-freedom control architecture.

II. Plant Dynamics and Control Architecture

A. Plant Dynamics

A precision motion system is usually actuated by air bearings or magnetically levitated motors and thus friction-free. Such kind of system can be modeled as multi-mass-block model shown in Fig. 1, f is the system input (i.e., force), y is the system output (i.e., displacement). The transfer function of the plant model is

$$G_p(s) = \frac{1}{m}\left(\frac{1}{s^2} + \sum_{i=1}^{N-1}\frac{\gamma_i}{s^2 + 2\zeta_i\omega_i s + \omega_i^2}\right) \tag{1}$$

m is the total mass of the system, ω_i and ζ_i are the resonant frequency and damping ratio of the i-th resonant mode, respectively, γ_i is the model parameter determined by inherent vibration modes. Detailed discussions on the dynamics of the multi-mass-block model can be found in [30], [31].

B. Control Architecture

Two degree-of-freedom (DOF) control architecture shown in Fig. 2 is widely utilized in the motion control industry. F is the feedforward controller, G_c is the feedback controller, G_p is the plant model, r is the reference trajectory, y is the system output, e is the tracking error, u_{ff} is the feedforward control signal, u_{fb} is the feedback control signal, u_{ILC} is the ILC signal. The tracking error without ILC can be expressed as

$$e = \frac{1 - FG_p}{1 + G_cG_p}r. \tag{2}$$

Apparently, the ideal feedforward controller is

$$F = G_p^{-1}. \tag{3}$$

Therefore, the design principle of feedforward controller is to approximate the plant inversion at the frequency range of interest as much as possible. For a digital control system in practice, time delay is inevitable due to mechanical and electrical factors, the zero-order hold sampling will also introduce time delay of around a half sampling period. For a precision motion system requiring nano-level motion accuracy, even time delay of several hundred micron seconds, if not compensated, will lead to a big tracking error [8], [29]. Therefore, the effect of time delay and zero-order hold sampling must be considered. Then the plant model should be revised as

$$G_p(s) = \mathrm{e}^{-\tau s}\frac{1 - \mathrm{e}^{-Ts}}{Ts}\frac{1}{m}\left(\frac{1}{s^2} + \sum_{i=1}^{N-1}\frac{\gamma_i}{s^2 + 2\zeta_i\omega_i s + \omega_i^2}\right) \tag{4}$$

$\mathrm{e}^{-\tau s}$ is the transfer function of time delay part, τ is the time delay, T is the sampling period. Note that the italicized e in (2) denotes tracking error while the non-italicized e in (4) denotes the Euler's number, i.e., $\mathrm{e} = 2.718\cdots$. It can be proved that the plant inversion can be expressed as

$$G_p^{-1}(s) = ms^2 + m\left(\tau + \frac{T}{2}\right)s^3 + a_1s^4 + \cdots \tag{5}$$

see [8] for the detailed deductions. The feedforward controller compensates for the terms equal or lower than fourth order is

$$F(s) = m_as^2 + m_js^3 + m_ss^4 \tag{6}$$

This is the widely used acceleration, jerk and snap feedforward [2]–[4], [8], [29], [32]. Observing (5) and (6), the ideal acceleration feedforward coefficient is $m_a = m$, the ideal jerk feedforward coefficient is $m_j = m\left(\tau + \frac{T}{2}\right)$, the ideal snap feedforward coefficient is $m_s = a_1$. The acceleration feedforward compensates for the rigid body dynamics, the jerk feedforward compensates for the time delay at the low frequency range, and the snap feedforward compensates for part of the resonant dynamics and part of the time delay at the higher frequency range. The feedforward controller requires careful tuning to achieve optimal tracking performance, see [8], [29]. However, even the accurate feedforward parameters are attained, perfect tracking is impossible, since (6) is just an approximation of the plant inversion at the low frequency range while a reference trajectory with large derivatives contains non-negligible high frequency components. Therefore, an extra compensation feedforward part should be used in cooperation with acceleration, jerk and snap feedforward to get better plant inversion and thus better tracking performance. From the perspective of discrete time domain, the extra compensation part can be either an FIR or IIR filter, the design and tuning procedures of the extra compensation will be given in section V-B.

Remark 1. The feedforward controller $F(s)$ is non-causal. In frequency domain, feedforward control signal is $u_{ff}(s) = F(s)r(s)$, in time domain, we have $u_{ff}(t) = m_ar^{(2)}(t) + m_jr^{(3)}(t) + m_sr^{(4)}(t)$. Since trajectory $r(t)$ and its derivatives are known in advance, the non-causality of $F(s)$ is not a problem for implementation.

III. Iterative Learning Control

Iterative learning control (ILC) is the most powerful feedforward strategy in practice. Various kinds of ILC algorithms

exist such as PD-type ILC, $\mathcal{H}_\infty$ ILC and model inversion ILC, see [18]. In the motion control industry, model inversion ILC (MIILC) is widely utilized, where the learning law is the inversion of corresponding process model.

The standard ILC can be expressed as

$$u_{\text{ILC}}^{(k+1)}(s) = Q(s)\left(u_{\text{ILC}}^{(k)}(s) + L(s)e^{(k)}(s)\right) \tag{7}$$

The superscript (k) denotes k-th iteration, s denotes the Laplace variable, $Q(s)$ is the transfer function of Q-filter, $L(s)$ is transfer function of the learning law, $u_{\text{ILC}}(s)$ is the Laplace transform of the ILC signal, $e(s)$ is the Laplace transform of the tracking error. Note that (7) is the expression of ILC in frequency domain, ILC can be also expressed in time domain such as the lifted-form ILC, see [18] for detailed discussions on the different forms of ILC. For ease of notation, the Laplace variable s is eliminated in the subsequent analysis.

It is not hard to get the tracking error after utilization of $u_{\text{ILC}}^{(k+1)}$

$$e^{(k+1)} = e^{(k)} - S_i\left(u_{\text{ILC}}^{(k+1)} - u_{\text{ILC}}^{(k)}\right) \tag{8}$$

Where $S_i = \frac{G_p}{1+G_cG_p}$ is the input disturbance sensitivity function. Assume $Q = 1$, if the learning law is the inverse of S_i, i.e., $L = S_i^{-1}$, then $e^{(k+1)} = 0$, in other words, ILC converges after only one iteration. Since this kind of ILC utilizes the inverse of corresponding model, it is called model-inversion ILC (MIILC). In precision motion control industry where high quality nominal model is not hard to attain, MIILC can usually provide very high performance (i.e., high convergence speed and high tracking performance) and is thus widely utilized. Note that $u_{fb} = G_c e$ and $S_i^{-1}e = T_c^{-1}u_{fb}$. u_{fb} is the feedback control signal, $T_c = \frac{G_cG_p}{1+G_cG_p}$ is the complementary sensitivity function. In this paper, we use $T_c^{-1}u_{fb}$ to generate ILC signal.

Due to the time delay and zero-order hold sampling, non-minimum-phase zeros may well occur [14]–[16], [33], making direct inversion of S_i or T_c infeasible. Therefore, stable-inversion methods such as ZPETC, ZMETC and NMPZITC have to be utilized [7], [14]–[16]. In this paper, NMPZITC is used to generate ILC signal.

IV. Simulation Parameters

The next section will be accompanied with simulation results, simulation parameters will thus be introduced here first. The non-collocated double-mass-block model shown in Fig. 3 is used as the plant model, the transfer function is

$$G_{p_0}(s) = \frac{1}{(m_1+m_2)}\left(\frac{1}{s^2} - \frac{1}{s^2+2\zeta_n\omega_n s+\omega_n^2}\right). \tag{9}$$

f is the system input (i.e., force), y is the system output (i.e., displacement). The model parameters are listed in Table I. Time delay $\tau = 2.5T$ is added to the plant, $T = 200\,\mu\text{s}$ is the sampling period.

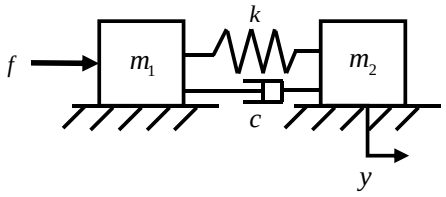


Fig. 3. Non-collocated double-mass-block model.

TABLE I
THE MODEL PARAMETERS USED IN SIMULATION

m_1	m_2	ω_n	ζ_n
5 kg	20 kg	700 Hz	0.03

With time delay and zero-order hold sampling considered, the equivalent plant model is $G_p(s) = \mathrm{e}^{-\tau s}\frac{1-\mathrm{e}^{-Ts}}{Ts}G_{p_0}(s)$. Expanding the inversion of $G_p(s)$ at $s = 0$, we can get

$$G_p^{-1}(s) = (m_1+m_2)s^2 + \frac{1}{2}(T+2\tau)(m_1+m_2)s^3 + \frac{(m_1+m_2)\left(12+T^2\omega_n^2+6T\tau\omega_n^2+6\tau^2\omega_n^2\right)s^4}{12\omega_n^2} + \cdots \tag{10}$$

In the simulation, the time delay is approximated by second order Pade approximation

$$\mathrm{e}^{-\tau s} \approx \frac{1-\frac{s\tau}{2}+\frac{s^2\tau^2}{12}}{1+\frac{s\tau}{2}+\frac{s^2\tau^2}{12}}. \tag{11}$$

Transferring the plant into discrete time domain with zero-order hold sampling method, we can get the discrete plant model. PI + double lead controller is used as feedback controller

$$G_c(z) = K\left(\frac{z+z_1}{z-1}\right)\left(\frac{z+z_2}{z+p_2}\right)\left(\frac{z+z_3}{z+p_3}\right). \tag{12}$$

The control bandwidth (the crossover frequency at $0\,\text{dB}$ in the Bode diagram of the open loop transfer function) is

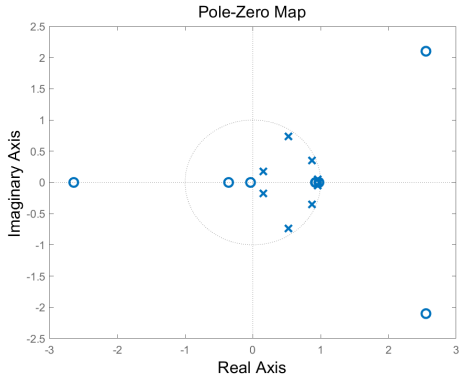


Fig. 4. The pole-zero map of the complementary sensitivity function.

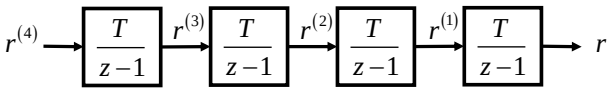
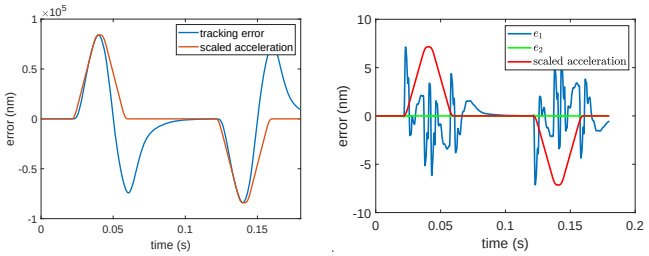


Fig. 5. The generation of reference trajectory.

TABLE II
THE TRAJECTORY PARAMETERS USED IN SIMULATION

trajectory	dis (mm)	vel (mm/s)	acc (m/s^2)	jerk (m/s^3)	snap (m/s^4)
trajectory one	60	600	30	2000	6×10^6
trajectory two	60	700	25	2500	6×10^6



(a) (b)

Fig. 6. (a) The tracking error with no feedforward. (b) e_1: the tracking error with tuned accleration, jerk and snap feedforward; e_2: the tracking error with tuned accleration, jerk, snap feedforward and ILC.

180 Hz. Fig. 4 gives the pole-zero map of the complementary sensitivity function, non-minimum phase zeros can be clearly observed. Fourth order trajectory is used as reference trajectory, the detailed trajectory planning algorithm can be found in [32]. *dis* stands for displacement, *vel*, *acc*, *jerk* and *snap* stand for the first to fourth order derivatives. Unless otherwise stated, trajectory one given in Table II is used. In implementation, the reference trajectory and its derivatives are attained by integrating from snap, see Fig. 5. Note that every backward integration will introduce around a half sampling period time delay, then the planned acceleration is around one sampling period ahead than the reference trajectory, etc. Therefore, according to (10), the ideal acceleration feedforward coefficient is $m_a = m_1 + m_2 = 25\,\mathrm{kg}$, the ideal jerk feedforward coefficient is $m_j = m\left(\tau + \frac{T}{2} - T\right) = 0.01\,\mathrm{kg\,s}$, the ideal snap feedforward coefficient is $m_s = \frac{(m_1+m_2)\left(12+T^2\omega_n^2+6T\hat{\tau}\omega_n^2+6\hat{\tau}^2\omega_n^2\right)}{12\omega_n^2} = 3.25 \times 10^{-6}\,\mathrm{kg\,s^2}$, where $\hat{\tau} = \tau - T = 1.5T$. The $-T$ in $m_j = m\left(\tau + \frac{T}{2} - T\right)$ and $\hat{\tau} = \tau - T$ comes from the fact that one sampling period time delay has been compensated by acceleration feedforward since the planned acceleration signal is one sampling period ahead compared with the reference trajectory.

V. FEEDFORWARD TUNING ALGORITHM

A. Tuning of acceleration, jerk and snap feedforward

The first step is the tuning of acceleration, jerk and snap feedforward. Setting feedforward controller $F = 0$, Fig. 6 (a) gives the tracking performance with no feedforward. Then use ILC to improve tracking performance, in the simulation example here, ILC converges after 2 iterations. With the converged ILC signal, almost perfect tracking control can be attained, then the ILC signal can be regarded as the ideal feedforward signal. Acceleration, jerk and snap feedforward contribute to the main part of the ideal feedforward control signal, then we can tune the feedforward coefficients by solving the following least squares problem

$$\min \|Ax - b\|_2 \tag{13}$$

$x = \begin{pmatrix} m_a & m_j & m_s \end{pmatrix}^T$ is the feedforward coefficients vector, $b = \begin{pmatrix} u_{\mathrm{ILC}}(1) & \cdots & u_{\mathrm{ILC}}(N) \end{pmatrix}^T$ is the ILC signal vector, N is the number of time instances used for tuning, A is data matrix

$$A = \begin{pmatrix} r^{(2)}(1) & r^{(3)}(1) & r^{(4)}(1) \\ \vdots & \vdots & \vdots \\ r^{(2)}(N) & r^{(3)}(N) & r^{(4)}(N) \end{pmatrix} \tag{14}$$

$r^{(i)}$ denotes the i-th order derivative of the reference trajectory. Then we can get

$$x = \left(A^T A\right)^{-1} A^T b. \tag{15}$$

The blue line in Fig. 6 gives the tracking error under tuned acceleration, jerk and snap feedforward. The tuned acceleration, jerk and snap coefficients are $25\,\mathrm{kg}$, $0.01\,\mathrm{kg\,s}$, $2.16 \times 10^{-6}\,\mathrm{kg\,s^2}$, respectively.

Remark 2. Compared with the theoretical values given in section IV, it can be seen that accurate acceleration and jerk feedforward coefficients can be found, and there is some gap between the tuned snap feedforward coefficient and the theoretical value. This is because: 1) time delay is approximated by second order Pade approximation in the simulation, this approximation contributes to part of the gap; 2) acceleration, jerk and snap feedforward coefficients are tuned by fitting ILC signal, the ILC signal contains not only control effort contributed by acceleration + jerk + snap feedforward, but also control effort contributed by higher order feedforward terms, this factor also contributes to part of the gap.

B. Tuning of the Extra Compensation Part

The second step is the tuning of the extra compensation part. As shown in Fig. 6 (b), since acceleration, jerk and snap feedforward is just an approximation of the plant inversion at the low frequency range, residual tracking error is inevitable. For precision motion systems requiring nano-level motion accuracy, such kind of residual tracking error must be compensated and thus an extra compensation part is required, see ΔF in Fig. 7. Here, the input of ΔF is snap, see section VII-A for the reason of this choice.

The design and tuning of ΔF can be stated as follows:

1) First, use ILC to compensate the residual tracking error. The green line in Fig. 6 gives the tracking performance with tuned acceleration, jerk, snap feedforward and ILC. Since the residual tracking error after use of tuned acceleration + jerk + snap feedforward can be compensated by ILC, we expect that

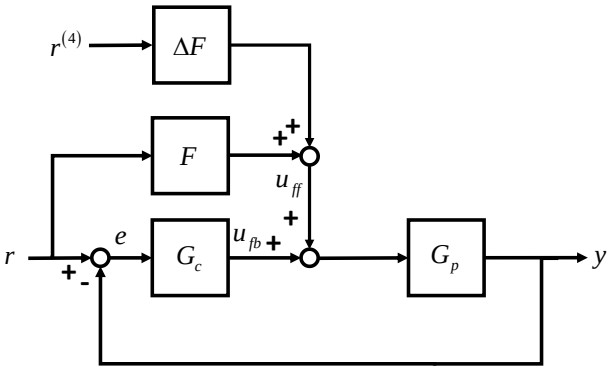


Fig. 7. Control architecture with extra compensation, ΔF is the model-based extra compensation part.

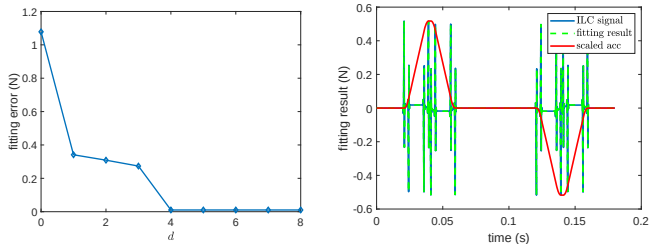


Fig. 8. (a) The smallest fitting errors with different d. (b) The fitting result when $d = 4$.

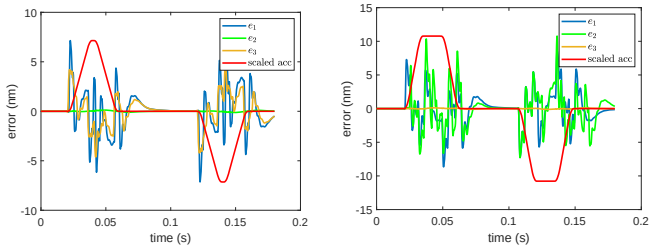


Fig. 9. (a) Tracking error under trajectory one, e_1: the tracking error with tuned acceleration, jerk and snap feedforward; e_2: the tracking error with tuned acceleration, jerk, snap feedforward and ΔF under $d = 4$; e_3: the tracking error with tuned acceleration, jerk, snap feedforward and ΔF under $d = 0$. (b) Tracking error under trajectory two, e_1: the tracking error with tuned accleration, jerk and snap feedforward; e_2: the tracking error with tuned accleration, jerk, snap feedforward and ILC which is learned under trajectory one; e_3: the tracking error with tuned acceleration, jerk, snap feedforward and ΔF under $d = 4$.

the output of the extra compensation part should approximate the ILC signal as much as possible.

2) Then choose a model-based feedforward controller

$$\Delta F(z) = z^d \frac{b_0 + b_1 z^{-1} + \cdots + b_m z^{-m}}{1 + a_1 z^{-1} + \cdots + a_n z^{-n}} = z^d G(z) \quad (16)$$

$G(z)$ is the causal part of $\Delta F(z)$, z^d is the forward shifting part of $\Delta F(z)$, which makes $\Delta F(z)$ non-causal, see section VII-B for the reasons for the utilization of

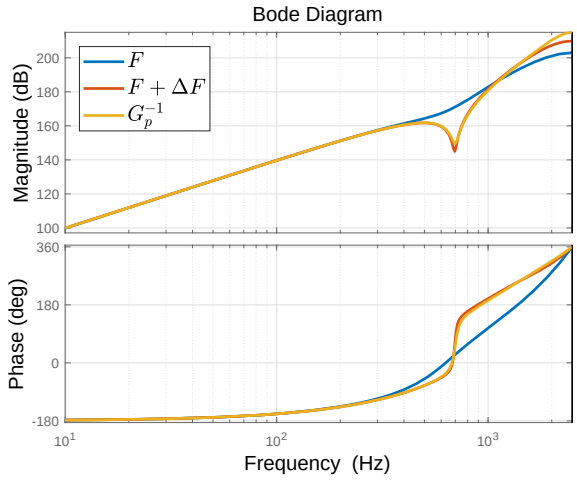


Fig. 10. The Bode plot of the plant inversion and feedforward controllers.

z^d in $\Delta F(z)$. Note that the non-causality of $\Delta F(z)$ is not a problem for implementation, since reference trajectory and its derivatives are known in advance. The number of forward steps can be determined by trial. The tuning of $\Delta F(z)$ is realized by approximating the ILC signal with the output of $\Delta F(z)$. We can attain the coefficients of $\Delta F(z)$ by the least squares algorithm (15), the data matrix A is $A = \begin{pmatrix} \varphi(t_0) & \cdots & \varphi(t_0 + N) \end{pmatrix}^T$, where $\varphi(i) = \begin{pmatrix} \varphi_{u_{\mathrm{ILC}}}(i) & \varphi_s(i) \end{pmatrix}^T$, $\varphi_{u_{\mathrm{ILC}}}(i) = \begin{pmatrix} -u_{\mathrm{ILC}}(i-1) & \cdots & -u_{\mathrm{ILC}}(i-n) \end{pmatrix}$, and $\varphi_s(i) = \begin{pmatrix} r^{(4)}(d+i) & \cdots & r^{(4)}(d+i-m) \end{pmatrix}$. This approach is also called the prediction error method from the perspective of system identification, see [34] for detailed discussions.

3) The choice of $\Delta F(z)$ is determined by Monte Carlo search, i.e., fix d, then search every possible combination of m and n within the given range; for every pair (m, n) , use the prediction error method to fit the ILC signal and get ΔF; find the ΔF with the smallest fitting error and choose the ΔF as the extra compensation part; change d and repeat the previous procedures until a satisfactory fitting result is attained.

In the simulation example here, the parameter space for searching is $\Omega = \{(m, n) \,|m \in [0, 20], m \in \mathbb{Z}, n \in [0, 20], n \in \mathbb{Z}\}$. Fig. 8 (a) gives the smallest fitting errors under different d. It shows that the best fitting is attained when $d \geq 4$. The reason why the fitting result remains unchanged when d is larger than a certain value can be found in section VII-B. Fig. 8 (b) gives the fitted signal when $d = 4$. When $d = 4$, the best result is given by a stable IIR filter with $m = 9$ and $n = 2$, the fitting error is $0.0103\,\mathrm{N}$. If ΔF is unstable, further processing is required, see section VII-C. Fig. 9 (a) gives the tracking performance with the extra compensation part, it can be seen that the residual tracking error and settling time is greatly attenuated. While the green line in Fig. 9 (a) gives the tracking error with ΔF under $d = 4$, the yellow line gives the tracking error with ΔF under $d = 0$, it can be seen that the performance of the extra compensation is influenced by non-causality. Fig. 9 (b) gives the tracking

Fig. 11. Experiment platform: a pneumatic motion stage actuated by linear motor.

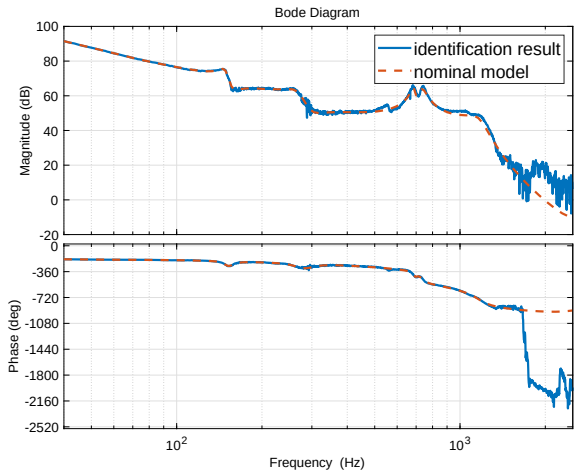


Fig. 12. System identification.

performance under trajectory two, it can be seen that while ILC is sensitive to trajectory variation (the green line in Fig. 9 (b), the ILC signal is learned under trajectory one), the extra compensation is trajectory invariant (the yellow line in Fig. 9 (b)), since it is model-based. Fig. 10 gives the Bode plot of the plant inversion, it can been that with the extra compensation, better plant inversion can be attained.

VI. Experiment

A. Experiment Platform

Experiment is carried on the precision motion system shown in Fig. 11, the system is a pneumatic motion stage actuated by linear motor. Displacement is measured by laser interferometer with $0.15\,\mathrm{nm}$ resolution, the sampling period is $T = 200\,\mu\mathrm{s}$. Fig. 12 gives the system identification result, the $-40\,\mathrm{dB/dec}$ slope at the low frequency range denotes the rigid body dynamics, several resonant modes occur at higher frequency range. The phase lag in the phase plot shows the very existence of time delay, time delay is around $600\,\mu\mathrm{s}$. PID is used as feedback controller, the control bandwidth is $70\,\mathrm{Hz}$ with 33.5° phase margin. Trajectory three and four given in Table III are used in the experiment.

TABLE III
The trajectory parameters used in experiment

trajectory	dis (mm)	vel (mm/s)	acc ($\mathrm{m/s^2}$)	jerk ($\mathrm{m/s^3}$)	snap ($\mathrm{m/s^4}$)
trajectory three	300	500	20	1500	2×10^5
trajectory four	300	400	16	1260	1×10^5

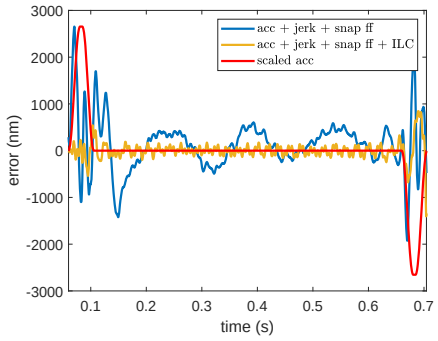


Fig. 13. Tracking performance under trajectory three with tuned acceleration + jerk + snap feedforward and ILC.

B. Experimental Results

The blue line in Fig. 13 gives the tracking error under trajectory three with tuned acceleration + jerk + snap feedforward. Then use ILC to further compensate the residual tracking error, the ILC converges after 3 iterations. The yellow line in Fig. 13 gives the tracking error with tuned acceleration + jerk + snap + ILC feedforward. The ILC signal is utilized to tune ΔF, the parameter space $\Omega = \{(m,n)\,|m \in [0,20]\,, m \in \mathbb{Z}, n \in [0,20]\,, n \in \mathbb{Z}\}$ is searched, Fig. 14 (a) gives the optimal fitting error under different d, the effect of non-causality can be clearly observed. Fig. 14 (b) shows the optimal fitting result with $d = 9$, the corresponding ΔF is a stable IIR filter with $m = 20$ and $n = 1$.

Fig. 15 gives the tracking performance with ΔF, it can be seen that with the extra compensation, tracking performance is enhanced and settling time is attenuated.

Moving average (MA)

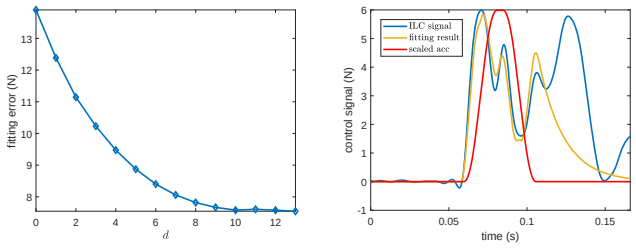
$$\mathrm{MA}\,(k) = \frac{1}{N} \sum_{i=k-\frac{N}{2}}^{k+\frac{N}{2}} e\,(i) \tag{17}$$

and moving standard deviation (MSD)

$$\mathrm{MSD}\,(k) = \sqrt{\frac{1}{N} \sum_{i=k-\frac{N}{2}}^{k+\frac{N}{2}} \left(e\,(i) - \mathrm{MA}\,(k)\right)^2} \tag{18}$$

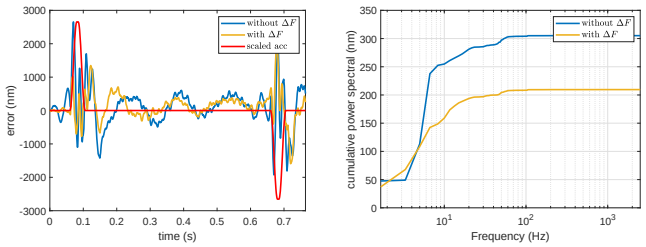
are two key performance indexes in precision motion control, N is the window size. Here, we use $N = 210$. Fig. 16 gives the MA and MSD with ΔF, the improvement of tracking performance can be clearly observed.

Fig. 17 gives the tracking performance under trajectory four, it can be seen that, since ΔF is model-based, the enhancement of tracking performance is insensitive to trajectory variation.



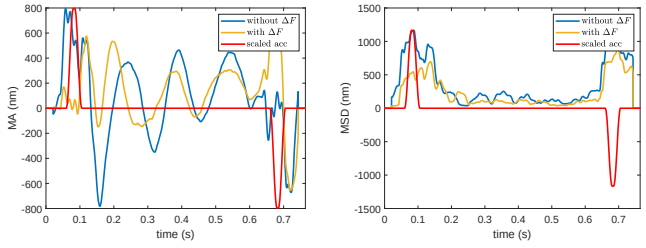
(a) (b)

Fig. 14. (a) The smallest fitting errors with different d. (b) The fitting result when $d = 9$.



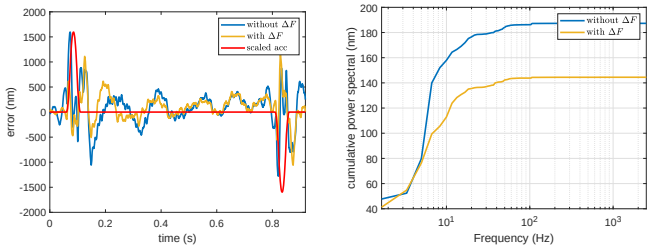
(a) (b)

Fig. 15. Tracking performance under trajectory three: (a) tracking error. (b) cumulative power spectral of tracking error.



(a) (b)

Fig. 16. Tracking performance under trajectory three: (a) MA of tracking error. (b) MSD of tracking error.



(a) (b)

Fig. 17. Tracking performance under trajectory four: (a) tracking error. (b) cumulative power spectral of tracking error.

VII. Further Discussions

A. The Input of ΔF

In simulation, the input of ΔF is chosen as the snap signal, the reason for this choice is that the dynamics equal or lower than fourth order can be well compensated by well-tuned acceleration + jerk + snap feedforward. The input signal can also be chosen as other lower order derivatives, e.g., the acceleration of reference trajectory, since snap can be expressed as acceleration filtered by a second order difference operator which is also a transfer function.

In experiment, the input of ΔF is chosen as the acceleration signal. The reason is that, in practice, external disturbance is inevitable, the existence of disturbance will affect the tuning accuracy of acceleration + jerk + snap feedforward, aside from this, part of the disturbance can be possibly modeled as a signal attained by filtering signals such as acceleration, jerk and snap with a transfer function.

B. The Non-causality of ΔF

As shown in previous sections, the extra compensation $\Delta F(z)$ can provide better performance when it is non-causal. Here, we give quantitative analysis about this phenomenon. For simplification, instead of the non-collocated double-mass-block model used in simulation, rigid body model is utilized here. The plant model is

$$G_p(s) = \frac{1}{ms^2}. \tag{19}$$

Transferring the plant into discrete time domain with zero-order hold sampling, we can get

$$G_p(z) = \frac{\frac{1}{2}(1+z)}{\frac{m}{T^2}(1-z)^2} \tag{20}$$

T is the sampling period, m is the mass of the system. By Taylor expansion at $z = 1$ (expansion at $s = 1$ in continuous time domain is equivalent to expansion at $z = 1$ in discrete time domain), we can get

$$\begin{aligned} G_p^{-1}(z) &= \frac{\frac{m}{T^2}(1-z)^2}{\frac{1}{2}(1+z)} \\ &= m\left(\frac{z-1}{T}\right)^2 + \left(-\frac{mT}{2}\right)\left(\frac{z-1}{T}\right)^3 \\ &+ \frac{mT^2}{4}\left(\frac{z-1}{T}\right)^4 + O(z-1)^5 \end{aligned} \tag{21}$$

$O(\cdot)$ denotes higher order terms. Note that $\frac{z-1}{T}$ is difference operator, then the ideal acceleration + jerk + snap feedforward is

$$F(z) = m\left(\frac{z-1}{T}\right)^2 - \frac{mT}{2}\left(\frac{z-1}{T}\right)^3 + \frac{mT^2}{4}\left(\frac{z-1}{T}\right)^4. \tag{22}$$

The ideal feedforward controller is the inversion of plant model, i.e., $F_{\text{ideal}}(z) = G_p^{-1}(z)$. Therefore, after utilization of ideal acceleration + jerk + snap feedforward, we need an extra feedforward controller to realize perfect tracking, and the ideal extra feedforward controller is

$$\begin{aligned} \Delta F_{\text{ideal}}(z) &= G_p^{-1}(z) - F(z) = \frac{-m}{4T^2}\frac{(z-1)^5}{1+z} \\ &= \frac{-m}{4T^2}z^4\frac{\left(1-z^{-1}\right)^5}{1+z^{-1}} \end{aligned} \tag{23}$$

which is a non-causal IIR filter with relative order 4. (23) can also be expressed as

$$\Delta F_{\text{ideal}}(z) = \frac{-m}{4T^2} z^{(4+p)} \frac{z^{-p}\left(1-z^{-1}\right)^5}{1+z^{-1}} \tag{24}$$

where p is a non-negative integer. According to (16), $\frac{z^{-p}\left(1-z^{-1}\right)^5}{1+z^{-1}}$ can still be well approximated by $G(z)$, since both $\frac{z^{-p}\left(1-z^{-1}\right)^5}{1+z^{-1}}$ and $G(z)$ are causal. That is why in Fig. 8 (a) and Fig. 14 (a), the fitting error remains unchanged when d is larger than a certain threshold. However, if p is negative, $\frac{z^{-p}\left(1-z^{-1}\right)^5}{1+z^{-1}}$ becomes non-causal and cannot be well approximated by a causal filter (i.e., $G(z)$). This mismatch will lead to the performance deterioration of the extra compensation. That is why in Fig. 8 (a) and Fig. 14 (a), the fitting performance deteriorates when the forward order of ΔF is not enough.

C. The Stability of ΔF

As mentioned before, ΔF is determined by Monte Carlo search and can be either an FIR or IIR filter. If ΔF is an FIR filter, there is no stability problem, if ΔF is an IIR filter and has unstable poles, stable approximation techniques such as ZPETC, ZMETC and NMPZITC have to be utilized [14]–[16]. The approximation capability of these methods can be further improved by adding difference operators, see [7], [35] for detailed discussions.

D. Comparisons with ILC

The proposed approach utilizes ILC but is not ILC. ILC is data-based while the proposed method is model-based. For repetitive process, ILC is very effective to attenuate repetitive tracking error. The main drawback of ILC is its sensitivity to trajectory variation, i.e., once the reference trajectory is changed, ILC has to be relearned, which is not a problem for model-based feedforward controllers. Therefore, model-based feedforward is still a very important technique and utilized in most motion systems in practice. By fitting ILC signal with a model-based feedforward controller, we can maintain the advantage of model-based feedforward while approximate the performance of ILC to some extent.

However, ILC still has its own advantages. ILC can attenuate both reference trajectory induced tracking error and tracking error caused by repetitive external disturbance, but model-based feedforward controllers can only attenuate reference trajectory induced tracking error and is unable to handle disturbance induced tracking error (unless the model of external disturbance is known, which is usually hard to attain in many applications). Aside from this, ILC applies for both linear and nonlinear systems. For nonlinear systems which cannot be described by transfer functions, the proposed model-based method does not work. High performance precision motion systems are usually designed to be with good linearity and high stiffness, making the proposed approach feasible. However, for systems with marked nonlinearity, ILC is still a better candidate compared with the proposed method.

On the whole, the idea of the proposed approach is to utilize ILC for the design and tuning of model-based feedforward controller, but ILC still has its special advantages over model-based strategies for the moment.

E. Comparisons with Previous Model-based Feedforward Strategies

The proposed approach adopts acceleration + jerk + snap feedforward + extra compensation ΔF. Acceleration + jerk + snap feedforward is a well-established feedforward strategy in precision motion control [1], [3]–[5], [8], [10], [13], [17], [28], [29], [36]. Due to the limited capability of acceleration + jerk + snap feedforward to approximate plant inversion, how to realize further feedforward compensation is extensively studied in previous research.

In [1], [10], [13], the extra compensation ΔF is an FIR filter. The structure of ΔF is determined by trial-and-error. In [37], [38], the feedforward controller is a rational function (i.e., an IIR filter), the structure of the feedforward controller is determined by trial-and-error and prior knowledge on plant dynamics. Compared with the previous works, our approach has the following contributions:

1) The structure of feedforward controller determines its performance limit. For precision motion systems requiring nano-level motion accuracy, the structure of feedforward controller must be carefully designed. However, how to choose the structure of feedforward controller is still an open problem. In previous works, this is done by manual trial-and-error. In our approach, we determine the feedforward structure by Monte Carlo search.

2) From one perspective, Monte Carlo search is still a kind of trial-and-error method, but this method still has its own advantages. In previous works [1], [10], [13], [37]–[39], once the structure of feedforward controller is determined, data-based techniques are exploited to tune the feedforward parameters. Then the structure of the feedforward controller is changed and the tuning process is repeated, the whole procedure ends until a satisfactory tracking performance is attained. The total experimental cost of such kind of trial-and-error approach is not small. In our approach, once ILC signal is attained, the remaining design and tuning process is totally off-line. For precision motion systems, high quality nominal model is not hard to attain, therefore, model-inversion ILC usually converges very fast (in the simulation and experimental examples in this paper, model-inversion ILC converges after 2-3 iterations). The experimental cost of the proposed approach is thus relatively small compared with previous works.

Of course, the idea of tuning model-based feedforward controller by fitting ILC signal is also not new, see [28], [29]. The main distinctions of this paper with previous works can be stated as follows:

1) In [28], [29], FIR filter is utilized to fit ILC signal, in this paper, ΔF can be either an FIR or IIR filter according to the searching result. Compared with FIR filter, IIR filter can provide better fitting capability and smoother feedforward signal.

2) In [29], feedback control signal, which can be regarded as a special kind of ILC signal, is utilized to tune acceleration + jerk + snap feedforward. In this paper, aside from acceleration

+ jerk + snap feedforward, an extra compensation feedforward part is utilized for further performance improvement.

3) The non-causality of the extra compensation part is quantitatively analyzed which is not considered in previous research.

VIII. Conclusions

In this paper, a feedforward tuning algorithm by fitting iterative learning control signal for precision motion systems is proposed. Acceleration, jerk and snap feedforward are first tuned by fitting ILC signal with least squares algorithm. Then, to compensate the residual tracking error, an extra model-based compensation part is utilized. The extra compensation controller is determined by Monte Carlo search and tuned by fitting ILC signal. Since the extra compensation is model-based, it is insensitive to reference trajectory variation. Simulation and experiment on a precision motion system both validate the theoretical analysis and the proposed algorithm.

Acknowledgment

The authors thank Jintao Feng, Zhexin Su, Jiankun Zhang and Haisong Song for their crucial assistance in the experiment. The first author thanks Weicai Huang for the valuable discussion during the revision of this paper.

References

[1] M. Baggen, M. Heertjes, and R. Kamidi, "Data-based feed-forward control in MIMO motion systems," in *2008 American Control Conference*, DOI 10.1109/acc.2008.4586954. IEEE, Jun. 2008.

[2] M. Boerlage, "MIMO jerk derivative feedforward for motion systems," in *2006 American Control Conference*, DOI 10.1109/acc.2006.1657326. IEEE, 2006.

[3] M. Boerlage, R. Tousain, and M. Steinbuch, "Jerk derivative feedforward control for motion systems," in *American Control Conference, 2004. Proceedings of the 2004*, vol. 5, pp. 4843–4848. IEEE, 2004.

[4] H. Butler, "Position control in lithographic equipment [applications of control]," *IEEE Control Systems*, vol. 31, DOI 10.1109/mcs.2011.941882, no. 5, pp. 28–47, Oct. 2011.

[5] H. Butler, "Adaptive feedforward for a wafer stage in a lithographic tool," *IEEE Transactions on Control Systems Technology*, vol. 21, DOI 10.1109/tcst.2012.2188102, no. 3, pp. 875–881, May. 2013.

[6] G. M. Clayton, S. Tien, K. K. Leang, Q. Zou, and S. Devasia, "A review of feedforward control approaches in nanopositioning for high-speed SPM," *Journal of Dynamic Systems, Measurement, and Control*, vol. 131, DOI 10.1115/1.4000158, no. 6, p. 061101, 2009.

[7] L. Dai, L. Xin, Y. Zhu, and M. Zhang, "Quantitative tracking error analysis and feedforward compensation under different model-based feedforward controllers in different control architectures," *IEEE Transactions on Industrial Electronics*, DOI 10.1109/tie.2019.2960717, pp. 1–1, 2020.

[8] L. Dai, L. Xin, Y. Zhu, M. Zhang, and C. Hu, "The generation mechanism of tracking error during acceleration or deceleration phase in ultra-precision motion systems," *IEEE Transactions on Industrial Electronics*, DOI 10.1109/tie.2018.2878114, pp. 1–1, 2018.

[9] M. Heertjes and D. Bruijnen, "MIMO FIR feedforward design for zero error tracking control," in *2014 American Control Conference*, DOI 10.1109/acc.2014.6858752. IEEE, Jun. 2014.

[10] M. Heertjes, D. Hennekens, and M. Steinbuch, "MIMO feed-forward design in wafer scanners using a gradient approximation-based algorithm," *Control Engineering Practice*, vol. 18, DOI 10.1016/j.conengprac.2010.01.006, no. 5, pp. 495–506, May. 2010.

[11] M. Heertjes, R. Rampadarath, and R. Waiboer, "Nonlinear q-filter in the learning of nano-positioning motion systems," in *2009 European Control Conference (ECC)*, DOI 10.23919/ecc.2009.7074622. IEEE, Aug. 2009.

[12] M. Iwasaki, K. Seki, and Y. Maeda, "High-precision motion control techniques: A promising approach to improving motion performance," *IEEE Industrial Electronics Magazine*, vol. 6, DOI 10.1109/mie.2012.2182859, no. 1, pp. 32–40, Mar. 2012.

[13] Y. Jiang, Y. Zhu, K. Yang, C. Hu, and D. Yu, "A data-driven iterative decoupling feedforward control strategy with application to an ultra-precision motion stage," *IEEE Transactions on Industrial Electronics*, vol. 62, DOI 10.1109/TIE.2014.2327559, no. 1, pp. 620–627, 2015.

[14] J. A. Butterworth, L. Y. Pao, and D. Y. Abramovitch, "The effect of nonminimum-phase zero locations on the performance of feedforward model-inverse control techniques in discrete-time systems," in *2008 American Control Conference*, DOI 10.1109/acc.2008.4586900. IEEE, Jun. 2008.

[15] B. Rigney, L. Pao, and D. Lawrence, "Nonminimum phase dynamic inversion for settle time applications," *IEEE Transactions on Control Systems Technology*, vol. 17, DOI 10.1109/tcst.2008.2002035, no. 5, pp. 989–1005, Sep. 2009.

[16] M. Tomizuka, "Zero phase error tracking algorithm for digital control," *Journal of Dynamic Systems, Measurement, and Control*, vol. 109, DOI 10.1115/1.3143822, no. 1, p. 65, 1987.

[17] M. Heertjes, "Data-based motion control of wafer scanners," *IFAC-PapersOnLine*, vol. 49, DOI 10.1016/j.ifacol.2016.07.918, no. 13, pp. 1–12, 2016.

[18] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Systems*, vol. 26, DOI 10.1109/mcs.2006.1636313, no. 3, pp. 96–114, Jun. 2006.

[19] M. Heertjes and T. Tso, "Nonlinear iterative learning control with applications to lithographic machinery," *Control Engineering Practice*, vol. 15, DOI 10.1016/j.conengprac.2007.03.005, no. 12, pp. 1545–1555, Dec. 2007.

[20] I. Rotariu, M. Steinbuch, and R. Ellenbroek, "Adaptive iterative learning control for high precision motion systems," *IEEE Transactions on Control Systems Technology*, vol. 16, DOI 10.1109/tcst.2007.906319, no. 5, pp. 1075–1082, Sep. 2008.

[21] M. Sandipan, C. Joshua, and M. Tomizuka, "Precision positioning of wafer scanners segmented iterative learning control for nonrepetitive disturbances [applications of control]," *IEEE Control Systems*, vol. 27, DOI 10.1109/mcs.2007.384130, no. 4, pp. 20–25, Aug. 2007.

[22] D. Yu, Y. Zhu, K. Yang, C. Hu, and M. Li, "A time-varying q-filter design for iterative learning control with application to an ultra-precision dual-stage actuated wafer stage," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 228, DOI 10.1177/0959651814547443, no. 9, pp. 658–667, Aug. 2014.

[23] L. Blanken and T. Oomen, "Multivariable iterative learning control design procedures: From decentralized to centralized, illustrated on an industrial printer," *IEEE Transactions on Control Systems Technology*, vol. 28, DOI 10.1109/tcst.2019.2903021, no. 4, pp. 1534–1541, Jul. 2020.

[24] X. Ge, J. L. Stein, and T. Ersal, "Frequency-domain analysis of robust monotonic convergence of norm-optimal iterative learning control," *IEEE Transactions on Control Systems Technology*, vol. 26, DOI 10.1109/tcst.2017.2692729, no. 2, pp. 637–651, Mar. 2018.

[25] T. D. Son, G. Pipeleers, and J. Swevers, "Robust monotonic convergent iterative learning control," *IEEE Transactions on Automatic Control*, vol. 61, DOI 10.1109/tac.2015.2457785, no. 4, pp. 1063–1068, Apr. 2016.

[26] J. van Zundert and T. Oomen, "On inversion-based approaches for feedforward and ILC," *Mechatronics*, vol. 50, DOI 10.1016/j.mechatronics.2017.09.010, pp. 282–291, Apr. 2018.

[27] D. Yoon, X. Ge, and C. E. Okwudire, "Optimal inversion-based iterative learning control for overactuated systems," *IEEE Transactions on Control Systems Technology*, DOI 10.1109/tcst.2019.2917682, pp. 1–8, 2019.

[28] M. F. Heertjes and R. M. van de Molengraft, "Set-point variation in learning schemes with applications to wafer scanners," *Control Engineering Practice*, vol. 17, DOI 10.1016/j.conengprac.2008.08.004, no. 3, pp. 345–356, Mar. 2009.

[29] L. Dai, X. Li, Y. Zhu, and M. Zhang, "Auto-tuning of model-based feedforward controller by feedback control signal in ultraprecision motion systems," *Mechanical Systems and Signal Processing*, vol. 142, DOI 10.1016/j.ymssp.2020.106764, p. 106764, Aug. 2020.

[30] D. K. Miu, *Mechatronics: electromechanics and contromechanics*. Springer Science & Business Media, 2012.

[31] A. Preumont, *Vibration Control of Active Structures*. Springer Netherlands, 2011.

[32] P. Lambrechts, M. Boerlage, and M. Steinbuch, "Trajectory planning and feedforward design for electromechanical motion systems," *Control Engineering Practice*, vol. 13, DOI 10.1016/j.conengprac.2004.02.010, no. 2, pp. 145–157, Feb. 2005.

[33] K. Åström, P. Hagander, and J. Sternby, "Zeros of sampled systems," *Automatica*, vol. 20, DOI 10.1016/0005-1098(84)90062-1, no. 1, pp. 31–38, Jan. 1984.

[34] T. Söderström and P. Stoica, *System identification*. Prentice-Hall, Inc., 1988.

[35] L. Dai, X. Li, Y. Zhu, and M. Zhang, "Quantitative analysis on tracking error under different control architectures and feedforward methods," in *2019 American Control Conference (ACC)*, DOI 10.23919/acc.2019.8814413. IEEE, Jul. 2019.

[36] X. Li, K. Yang, Y. Zhu, and D. Yu, "Feedforward coefficient identification and nonlinear composite feedback control with applications to 3-DOF planar motor," *Journal of Mechanical Science and Technology*, vol. 27, DOI 10.1007/s12206-013-0105-z, no. 3, pp. 895–907, Mar. 2013.

[37] J. Bolder, J. van Zundert, S. Koekebakker, and T. Oomen, "Enhancing flatbed printer accuracy and throughput: Optimal rational feedforward controller tuning via iterative learning control," *IEEE Transactions on Industrial Electronics*, vol. 64, DOI 10.1109/tie.2016.2613498, no. 5, pp. 4207–4216, May. 2017.

[38] J. van Zundert, J. Bolder, and T. Oomen, "Optimality and flexibility in iterative learning control for varying tasks," *Automatica*, vol. 67, DOI 10.1016/j.automatica.2016.01.026, pp. 295–302, May. 2016.

[39] F. Boeren, D. Bruijnen, and T. Oomen, "Enhancing feedforward controller tuning via instrumental variables: with application to nanopositioning," *International Journal of Control*, vol. 90, DOI 10.1080/00207179.2016.1219921, no. 4, pp. 746–764, Sep. 2016.

[40] S. Skogestad and I. Postlethwaite, *Multivariable feedback control: analysis and design*, vol. 2. Wiley New York, 2007.

[41] D. Torfs, J. Swevers, and J. D. Schutter, "Quasi-perfect tracking control of non-minimal phase systems," in *[1991] Proceedings of the 30th IEEE Conference on Decision and Control*, DOI 10.1109/cdc.1991.261296. IEEE, 1991.

[42] D. Torfs, J. D. Schutter, and J. Swevers, "Extended bandwidth zero phase error tracking control of nonminimal phase systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 114, DOI 10.1115/1.2897354, no. 3, p. 347, 1992.

[43] X. Li, K. Yang, Y. Zhu, and D. Yu, "Non-linear composite control with applications to 3-DOF planar motor," *Transactions of the Institute of Measurement and Control*, vol. 35, DOI 10.1177/0142331212446040, no. 3, pp. 330–341, May. 2012.

[44] L. Dai, X. Li, Y. Zhu, and M. Zhang, "A high performance feedforward tuning approach for ultra-precision motion control," in *Proceedings of 32nd ASPE Annual Meeting*, pp. 439–444. ASPE, 2017.

[45] C.-H. Menq and J. jae Chen, "Precision tracking control of discrete time nonminimum-phase systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 115, DOI 10.1115/1.2899027, no. 2A, p. 238, 1993.

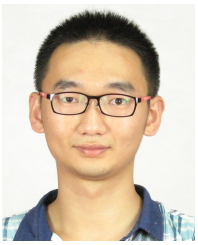
[46] Z. Hou, H. Gao, and F. L. Lewis, "Data-driven control and learning systems," *IEEE Transactions on Industrial Electronics*, vol. 64, DOI 10.1109/TIE.2017.2653767, no. 5, pp. 4070–4075, 2017.

[47] B. Haack and M. Tomizuka, "The effect of adding zeroes to feedforward controllers," *Journal of Dynamic Systems, Measurement, and Control*, vol. 113, DOI 10.1115/1.2896362, no. 1, p. 6, 1991.

[48] T.-C. Tsao and M. Tomizuka, "Adaptive zero phase error tracking algorithm for digital control," *Journal of Dynamic Systems, Measurement, and Control*, vol. 109, DOI 10.1115/1.3143866, no. 4, p. 349, 1987.

[49] H. Fujimoto, Y. Hori, and A. Kawamura, "Perfect tracking control based on multirate feedforward control with generalized sampling periods," *IEEE Transactions on Industrial Electronics*, vol. 48, DOI 10.1109/41.925591, no. 3, pp. 636–644, Jun. 2001.

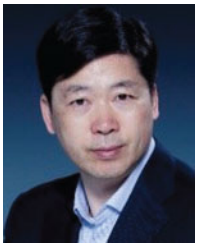
[50] T. Oomen and C. R. Rojas, "Sparse iterative learning control with application to a wafer stage: Achieving performance, resource efficiency, and task flexibility," *Mechatronics*, vol. 47, DOI 10.1016/j.mechatronics.2017.09.004, pp. 134–147, Nov. 2017.



Luyao Dai received B.S. degree in Mechanical Engineering from Tsinghua University, Beijing, China, in 2016. He is currently a Ph.D. candidate in Department of Mechanical Engineering, Tsinghua University. His reseach interest is ultra-precision motion control.

Xin Li received Ph.D. degree in Mechanical Engineering from Tsinghua University, Beijing, China, in 2013, M.S. and B.S. degrees in Mechanical Engineering from Dalian University of Technology, Dalian, China, in 2009 and 2006, respectively. His research interests include precision/ultra-precision motion control, robot control, especially the control and application of precision motion systems for engineering practice in industry.

He is currently an Assistant Professor in Department of Mechanical Engineering, Tsinghua University, Beijing. He has published more than 10 papers and applied for more than 10 invention patents.



Yu Zhu (M'12) received the B.S. degree in Radio Electronics from Beijing Normal University, Beijing, China, in 1983, and the M.S. degree in computer applications and the Ph.D. degree in Mechanical Design and Theory from China University of Mining and Technology, Beijing, China, in 1993 and 2001, respectively.

He is currently a Professor with the State Key Laboratory of Tribology, Department of Mechanical Engineering, Tsinghua University, Beijing, China. He has authored more than 180 technical papers. He holds more than 50 warranted invention patents. His research interests include precision measurement and motion control, ultra-precision mechanical design and manufacturing, two-photon micro-fabrication, and electronics manufacturing technology and equipment.

Ming Zhang received the B.S. degree in mechanical engineering from the University of Science and Technology Beijing, Beijing, China, in 1996 and the Ph.D. degree in precision instruments from Tsinghua University, Beijing, in 2005.

He is currently an Associate Professor with the Department of Mechanical Engineering, Tsinghua University. His research interests include ultraprecision mechanical design and manufacturing, system modeling and analysis, and design of high-precision linear motors and planar motors.