High-Precision Control of Ball-Screw-Driven Stage Based on Repetitive Control Using n-Times Learning Filter

Hiroshi Fujimoto, *Senior Member, IEEE*, and Tadashi Takemura, *Student Member, IEEE*

***Abstract*—This paper presents a novel learning control method for ball-screw-driven stages. In recent years, many types of friction models that are based on complicated equations have been studied. However, it is difficult to treat friction models with equations because the level of precision that is associated with real friction characteristics and parameter tuning are difficult to achieve. In contrast, repetitive perfect tracking control (RPTC) is a repetitive control technique that achieves high-precision positioning. In this paper, we propose the use of RPTC with n-times learning filter. The n-times learning filter has a sharper rolloff property than conventional learning filters. With the use of the n-times learning filter, the proposed RPTC can converge tracking errors n times faster than the RPTC with the conventional learning filter. Simulations and experiments with a ball-screw-driven stage show the fast convergence of the proposed RPTC. Finally, the proposed learning control scheme is combined with data-based friction compensation, and the effectiveness of this combination is verified for the x–y stage of a numerically controlled machine tool.**

***Index Terms*—n-times learning, perfect tracking control, repetitive control, zero-phase low-phase filter.**

I. Introduction

BALL-SCREW-DRIVEN stages are widely used in industrial equipment such as numerically controlled (NC) machine tools, and so on. From the viewpoint of productivity and microfabrication, high-speed and high-precision positioning techniques are desired from such types of industrial equipment. However, the high-precision positioning of ball-screw-driven stages is difficult because of their nonlinear friction characteristics [1], [2]. Therefore, various studies have been conducted on rolling friction compensation. Two types of rolling friction compensation methods were developed in these studies, i.e., the model-based compensation and the learning-based compensation (repetitive control or iterative learning control).

Under the model-based compensation, the generalized Maxwell-slip model [3], the variable natural length spring model [4], the rheology-based model [5], [6], and so on have been proposed. If designed well, these models can greatly suppress the influence of rolling friction. However, parameter determination is difficult in the construction of precise nonlinear fiction models.

In contrast, under the learning-based compensation [7], the influence of nonlinear friction can be suppressed by learning. This technique can be applied not only to ball-screw-driven stages but also to other devices because it does not require disturbance models [8]–[10]. Repetitive control (RC) is one of the learning control techniques that can achieve high-precision positioning. This technique is widely used for rejecting periodic disturbances or for tracking a periodic reference signal [11], [12]. As one of the RC techniques, repetitive perfect tracking control (RPTC) has been proposed in [13], and it achieved high tracking performance with a ball-screw-driven stage [14], [15]. From the standpoint of high-speed positioning, a small number of learning iterations are required.

Under ideal conditions, using the results of mathematical analysis, RPTC can suppress input (or output) disturbances and guarantee perfect tracking after a single learning iteration. Here, ideal conditions imply no modeling error and no use of a low-pass filter in the learning unit. However, modeling error is unavoidable and a low-pass filter is necessary for ensuring the robust stability of the control system.

In this paper, n-times learning RPTC is proposed. In the conventional RPTC, the high-speed convergence of learning is sacrificed for robust control system stability. In contrast, the proposed RPTC can ensure stability without sacrificing the high-speed convergence. Moreover, from a mathematical analysis, the tracking error of the proposed RPTC converges n times faster than the conventional RPTC with a small modeling error condition. In other words, the tracking error convergence that is achieved after n learning iterations using the conventional RPTC can be achieved with only one learning iteration using the proposed RPTC, as shown in Fig. 1. A simulation under the no-modeling-error condition shows that n-times faster error convergence is achieved by the proposed RPTC. In addition, the simulations and experiments with a ball-screw-driven stage indicate the effectiveness of the proposed RPTC.

II. Ball-Screw-Driven Stage

A. Experimental Stage

The experimental stage that is used in this paper is shown in Fig. 2. This ball-screw-driven stage is a part of an NC machine

Manuscript received October 9, 2012; revised January 6, 2013 and April 30, 2013; accepted June 14, 2013. Date of publication November 11, 2013; date of current version January 31, 2014.

The authors are with the Department of Advanced Energy, Graduate School of Frontier Sciences, The University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa, Chiba 277-8561, Japan (e-mail: hfuji@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2013.2290286

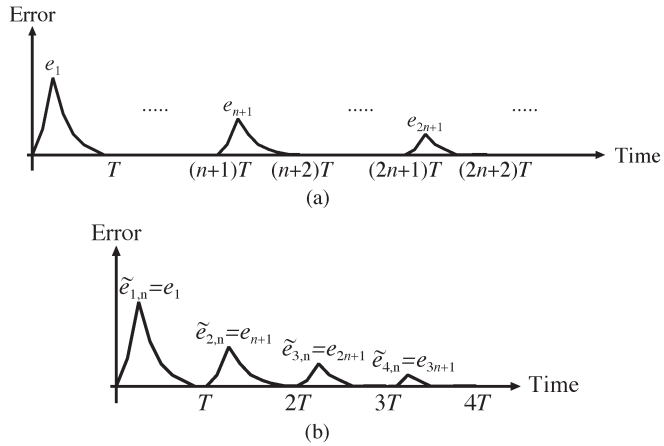

Fig. 1. Error convergence. (a) Conventional RPTC. (b) Proposed RPTC.

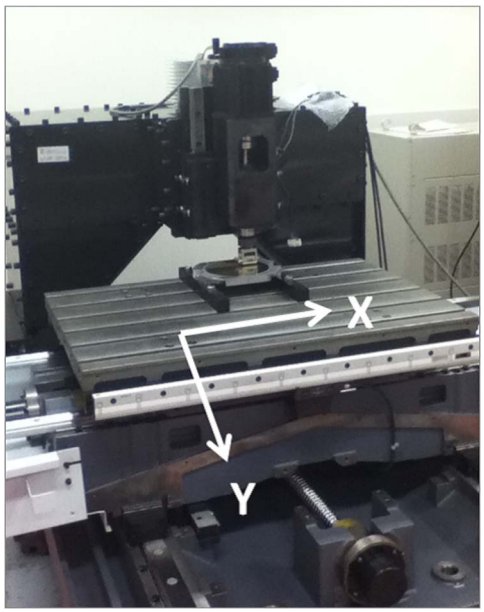

Fig. 2. Experimental stage.

tool product. The ball screw is directly coupled with the shaft of a servo motor. Each axis has a rotary encoder with a resolution of 5.99 μrad/pulse and a linear encoder with a resolution of 100 nm/pulse; these are attached to the servo motor and the table, respectively. Moreover, a grid encoder called KGM is installed to measure the contouring accuracy.

In this paper, only the rotary encoder is used in the position controller because we assume a semiclosed control, which is prevalent in commercial products. Furthermore, we only control the x-axis, except in Section VI-B.

Fig. 3 shows the frequency response that is obtained from current reference i_q^{ref} to the angular position using a fast Fourier transform analyzer. From this figure, nominal plant $P_n(s)$ is determined as follows:

$$P_n(s) = \frac{K_T}{Js^2 + Ds} \tag{1}$$

where the torque constant is $K_T = 0.715$ Nm/A, the inertia is $J = 0.01$ kgm^2, and the viscous friction is $D = 0.1$ Nms/rad.

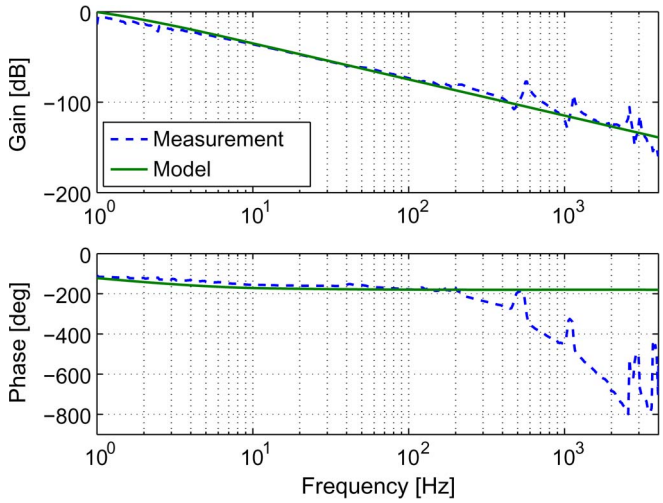

Fig. 3. Frequency response.

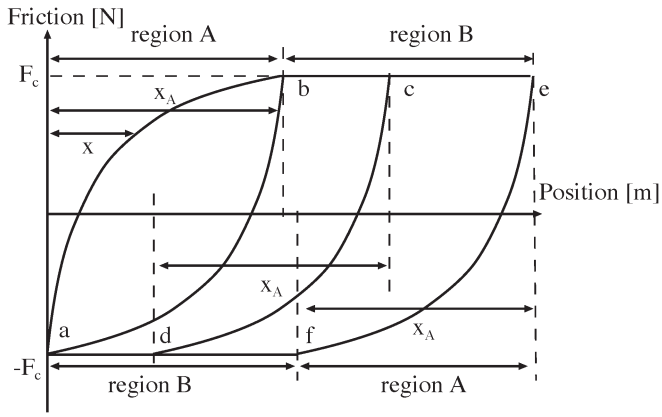

Fig. 4. Rolling friction characteristics.

B. Nonlinear Rolling Friction

Rolling friction is generated by the balls that are present in the ball screw and the rolling guide. The characteristics of this rolling friction are shown in Fig. 4. These characteristics can be divided into two regions, i.e., regions A and B. In region A, which is the start of the rolling region, the balls do not roll perfectly. At this time, the friction force depends on the displacement from the velocity reversal due to the balls' elastic deformation. In region B, which is the rolling region, the balls roll perfectly. At this time, the frictional property shows the Coulomb friction characteristic, and the frictional force attains a constant value.

As aforementioned, many mathematical models of this nonlinear friction have been already developed [3]–[6]. These equation-based models can greatly suppress the influence of rolling friction if a well-designed model is constructed. However, these models express the rolling friction characteristic using complex equations with various parameters. It is difficult to determine these parameters, and adequate know-how is required for parameter determination.

Therefore, in [14], a data-based friction model was developed. This friction model is based on the use of measured friction data and does not require any equations. However, both

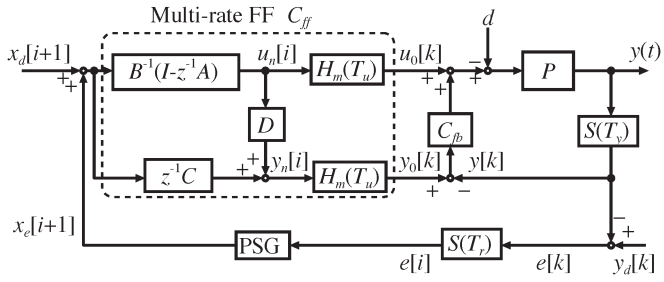

Fig. 5. RPTC block diagram.

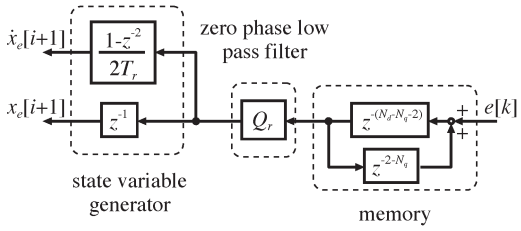

Fig. 6. PSG block diagram.

in the equation-based and data-based models, the robustness of the rolling friction characteristics is critical because the robustness varies with the temperature and the environment.

Furthermore, the position error of NC machine stages is generated not only by this friction but also by other disturbances, plant parameter variations, or mechanical resonance modes. Therefore, in this paper, learning control is considered to enhance the robustness of the servo mechanism.

III. Conventional RPTC

The RPTC is a repetitive control scheme that can iteratively (by learning) suppress periodic disturbances [13]. In this section, the structure of the RPTC and its convergence condition are explained.

A. RPTC Structure

A block diagram of the RPTC scheme is shown in Fig. 5. The RPTC consists of feedback (FB) controller $C_{\rm fb}$, multirate feedforward (FF) controller $C_{\rm ff}$, which is based on perfect tracking control (PTC) [16], and a learning unit, which is also known as a periodic signal generator (PSG). x_d denotes the state vector of the target position and target velocity. In the PTC, reference r, which is the input of $C_{\rm ff}$, is given by the preview trajectory as $r[i] = x_d[i]$. y, u, and d denote the position output, the control input, and the disturbance, respectively. $S(T_y)$ and $S(T_r)$ denote the samplers with output sampling period T_y and reference sampling period T_r, respectively. $H_m(T_u)$ denotes a multirate holder with control period T_u.

The PSG that is shown in Fig. 6 is composed of a memory module, learning filter (low-pass filter) Q_r, and a state-variable generator. The memory module memorizes tracking error e and compensates for the sample delays that occur in Q_r and in the state-variable generator. For the smoothing of FF input $u_0[k]$, the sensor noise in $y[k]$ should be attenuated by low-pass filter Q_r [13].

N_d is the number of memory units that is required in the RPTC. N_d is given as $N_d = T_d/T_r$, where T_d is a disturbance cycle, and T_r is the multirate sampling time of the reference

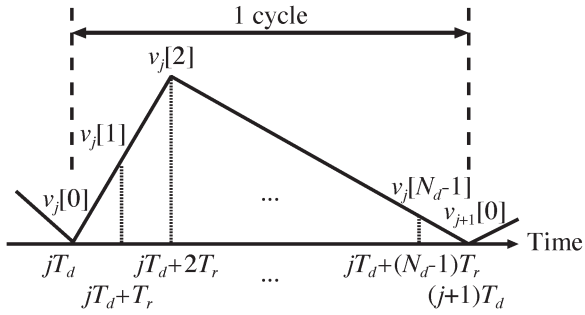

Fig. 7. Iterative signal.

trajectory, which is determined by the PTC [16]. Learning filter Q_r is a realization of a zero-phase low-pass filter Q. Q and Q_r are expressed by

$$Q = \left(\frac{z+2+z^{-1}}{4}\right)^{N_q} \tag{2}$$

$$Q_r = z^{-N_q} Q \tag{3}$$

where Q_r can be realized due to the sample delay compensation of the memory. The state-variable generator generates the state variables, the position, and the velocity, which is required by the multirate FF controller that is mentioned in the PTC theory. Velocity is calculated using the central difference for avoiding sample delay. Periodic disturbances are suppressed as follows.

- Step 1 (learning). The tracking error is stored in the memory.
- Step 2 (the state variable calculation). The state variable is calculated from the stored error.
- Step 3 (the redesigning of the target trajectory). The calculated state variable is added to the target trajectory. Then, the error becomes smaller than the previous cycle.

B. Definition of Signals and Transfer Functions

Signals in iteration cycle $T_d(= N_d T_r)$, as shown in Fig. 7, are expressed as the following vector:

$$\boldsymbol{v}_j = [v_j[0],\, v_j[1], \ldots, v_j[N_d-1]]^T. \tag{4}$$

Here, T_r and subscript j denote the sampling time and the number of iterations, respectively. A transfer function can be represented using impulse response matrix $\boldsymbol{G}$ [17], i.e.,

$$\boldsymbol{G} = \begin{bmatrix} g_0 & 0 & \ldots & 0 \\ g_1 & g_0 & \ldots & 0 \\ \vdots & \vdots & \ldots & \vdots \\ g_{N_d-1} & g_{N_d-2} & \ldots & g_0 \end{bmatrix}. \tag{5}$$

Here, $g_i (i = 1, 2, \ldots, N_d - 1)$ denotes the impulse response coefficient of transfer function G, which is described in Fig. 8. The input–output relation of each sampling time T_r in iteration cycle T_d is described using input–output vectors and impulse response matrices as follows:

$$\boldsymbol{G}_j = \boldsymbol{G}\boldsymbol{H}\boldsymbol{u}_j = \boldsymbol{H}\boldsymbol{G}\boldsymbol{u}_j \tag{6}$$

where $\boldsymbol{G}$ and $\boldsymbol{H}$ are arbitrary impulse response matrices. The multiplication of $\boldsymbol{G}$ and $\boldsymbol{H}$ is commutative because they are lower triangular and Toeplitz matrices.

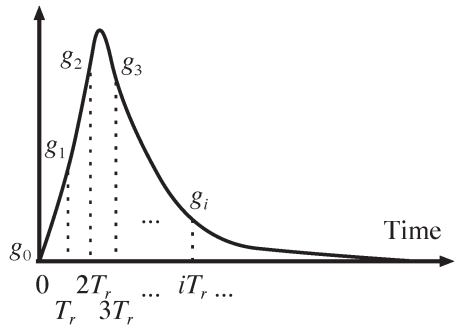

Fig. 8. Impulse response of G.

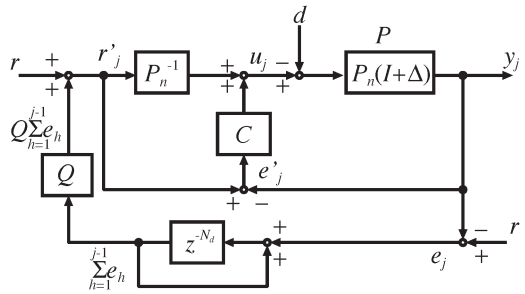

Fig. 9. Equivalent block diagram of the RPTC.

C. RPTC Convergence Condition

The RPTC convergence condition is derived using the equivalent block diagram that is shown in Fig. 9. This equivalent block diagram is based on the following assumptions.

- Usually, the RPTC is a multirate control system. However, here, the RPTC is treated as a single-rate system for simplicity.
- Multirate FF controller $C_{\rm ff}$ is treated as $C_{\rm ff} = P_n^{-1}$ because this is a perfect inverse system, in terms of both the gain and the phase of the plant.
- To simplify the system, only the position is used as the input to $C_{\rm ff}$, despite the fact that $C_{\rm ff}$ requires a state variable as its input. Owing to this assumption, the state-variable generator can be ignored.
- FB controller $C_{\rm fb}$ stabilizes plant P, i.e., $S=(1+PC_{\rm fb})^{-1}$ is stable.

The RPTC convergence condition is derived from the relationship between the error vectors of the jth and $(j+1)$th iterations. The output vector of the jth iteration, i.e., $\boldsymbol{y}_j$, is given as

$$\begin{aligned}\boldsymbol{y}_j &= \boldsymbol{P}(\boldsymbol{u}_j - \boldsymbol{d}) \\ &= \boldsymbol{P}\left\{\left(\boldsymbol{P}_n^{-1} + \boldsymbol{C}\right)\boldsymbol{r}'_j - \boldsymbol{C}\boldsymbol{y}_j - \boldsymbol{d}\right\}.\end{aligned}$$

Therefore, we obtain

$$\begin{aligned}\boldsymbol{y}_j &= (\boldsymbol{I} + \boldsymbol{P}\boldsymbol{C})^{-1}\boldsymbol{P}\left\{\left(\boldsymbol{P}_n^{-1} + \boldsymbol{C}\right)\boldsymbol{r}'_j - \boldsymbol{d}\right\} \\ &= \boldsymbol{S}\boldsymbol{P}\left\{\left(\boldsymbol{P}_n^{-1} + \boldsymbol{C}\right)\left(\boldsymbol{r} + \boldsymbol{Q}\sum_{h=1}^{j-1}\boldsymbol{e}_h\right) - \boldsymbol{d}\right\} \qquad (7)\end{aligned}$$

where

$$\boldsymbol{S} = (\boldsymbol{I} + \boldsymbol{P}\boldsymbol{C})^{-1}. \qquad (8)$$

From (7), the error vector of the jth iteration $\boldsymbol{e}_j$ is expressed as

$$\begin{aligned}\boldsymbol{e}_j &= \boldsymbol{r} - \boldsymbol{y}_j \\ &= \left\{\boldsymbol{I} - \boldsymbol{S}\boldsymbol{P}\left(\boldsymbol{P}_n^{-1} + \boldsymbol{C}\right)\right\}\boldsymbol{r} \\ &\quad - \boldsymbol{Q}\boldsymbol{S}\boldsymbol{P}\left(\boldsymbol{P}_n^{-1} + \boldsymbol{C}\right)\sum_{h=1}^{j-1}\boldsymbol{e}_h + \boldsymbol{S}\boldsymbol{P}\boldsymbol{d}. \qquad (9)\end{aligned}$$

The error vector of the next iteration $\boldsymbol{e}_{j+1}$ is expressed as

$$\begin{aligned}\boldsymbol{e}_{j+1} &= \left\{\boldsymbol{I} - \boldsymbol{S}\boldsymbol{P}\left(\boldsymbol{P}_n^{-1} + \boldsymbol{C}\right)\right\}\boldsymbol{r} - \boldsymbol{Q}\boldsymbol{S}\boldsymbol{P}\left(\boldsymbol{P}_n^{-1} + \boldsymbol{C}\right)\sum_{h=1}^{j}\boldsymbol{e}_h \\ &\quad + \boldsymbol{S}\boldsymbol{P}\boldsymbol{d} \\ &= \left(\boldsymbol{I} - \boldsymbol{Q}\boldsymbol{S}\boldsymbol{P}\boldsymbol{P}_n^{-1} - \boldsymbol{Q}\boldsymbol{S}\boldsymbol{P}\boldsymbol{C}\right)\boldsymbol{e}_j. \qquad (10)\end{aligned}$$

If plant $\boldsymbol{P}$ has a multiplicative modeling error $\boldsymbol{\Delta}$, then $\boldsymbol{P}$ is expressed as

$$\boldsymbol{P} = \boldsymbol{P}_n(\boldsymbol{I} + \boldsymbol{\Delta}). \qquad (11)$$

From (8), we have

$$\boldsymbol{S}\boldsymbol{P}\boldsymbol{C} = \boldsymbol{I} - \boldsymbol{S}. \qquad (12)$$

Then, $\boldsymbol{e}_{j+1}$ is written using (10)–(12) as follows:

$$\begin{aligned}\boldsymbol{e}_{j+1} &= \left\{\boldsymbol{I} - \boldsymbol{Q}\boldsymbol{S}(\boldsymbol{I} + \boldsymbol{\Delta}) - \boldsymbol{Q}(\boldsymbol{I} - \boldsymbol{S})\right\}\boldsymbol{e}_j \\ &= (\boldsymbol{I} - \boldsymbol{Q} - \boldsymbol{Q}\boldsymbol{S}\boldsymbol{\Delta})\boldsymbol{e}_j \\ &= (\boldsymbol{I} - \boldsymbol{Q} - \boldsymbol{Q}\boldsymbol{S}\boldsymbol{\Delta})^j\boldsymbol{e}_1. \qquad (13)\end{aligned}$$

Equation (13) indicates that error vector $\boldsymbol{e}_1$, which appeared in the first iteration, iteratively converges with ratio $(\boldsymbol{I} - \boldsymbol{Q} - \boldsymbol{Q}\boldsymbol{S}\boldsymbol{\Delta})$. Moreover, $\boldsymbol{e}_2 = \boldsymbol{0}$ (zero vector) under the ideal condition, which is given as $\boldsymbol{Q} = \boldsymbol{I}$, and $\Delta = \boldsymbol{O}$. This means that perfect tracking is achieved after the first learning, which is the most interesting property of the RPTC.

The design of Q is important because the stability and the error convergence depend on Q. The sufficient condition for monotonic convergence, which is expressed as

$$\|\boldsymbol{e}_1\|_2 > \|\boldsymbol{e}_2\|_2 > \cdots > \|\boldsymbol{e}_n\|_2 > \cdots \qquad (14)$$

is to satisfy the following inequality [17]:

$$\|\boldsymbol{I} - \boldsymbol{Q} - \boldsymbol{Q}\boldsymbol{S}\boldsymbol{\Delta}\|_\infty < 1. \qquad (15)$$

IV. Quick Convergence RPTC With n-Times Learning Filter $\tilde{Q}_n$ (Proposed)

Using the RPTC, the tracking error converges as the number of iterations increases. In this section, to improve the tracking error convergence, we propose the n-times learning RPTC scheme with an n-times learning filter $\tilde{Q}_n$. The proposed RPTC is realized by replacing learning filter Q with $\tilde{Q}_n$. Given that Q is replaced with $\tilde{Q}_n$, the convergence condition changes. Thus, when $\Delta = 0$, the proposed RPTC can learn n times faster than the conventional RPTC. The change in the convergence condition, owing to the learning filter replacement, is verified. Furthermore, the characteristics of $\tilde{Q}_n$ are described.

A. n-Times Learning Filter

When $\boldsymbol{e}_1$ occurs in the conventional RPTC, $\boldsymbol{e}_3,\ \boldsymbol{e}_4, \ldots, \boldsymbol{e}_{n+1}$ are written using the binomial theorem as follows:

$$\begin{aligned}\boldsymbol{e}_3 &= (\boldsymbol{I}-\boldsymbol{Q}-\boldsymbol{QS\Delta})^2\boldsymbol{e}_1 \\ &= \left\{(\boldsymbol{I}-\boldsymbol{Q})^2 - 2(\boldsymbol{I}-\boldsymbol{Q})\boldsymbol{QS\Delta} + (\boldsymbol{QS\Delta})^2\right\}\boldsymbol{e}_1 \quad (16)\\ \boldsymbol{e}_4 &= (\boldsymbol{I}-\boldsymbol{Q}-\boldsymbol{QS\Delta})^3\boldsymbol{e}_1 \\ &= \left\{(\boldsymbol{I}-\boldsymbol{Q})^3 - 3(\boldsymbol{I}-\boldsymbol{Q})^2\boldsymbol{QS\Delta}\right. \\ &\quad \left. + 3(\boldsymbol{I}-\boldsymbol{Q})(\boldsymbol{QS\Delta})^2 - (\boldsymbol{QS\Delta})^3\right\}\boldsymbol{e}_1 \quad (17)\\ \boldsymbol{e}_{n+1} &= (\boldsymbol{I}-\boldsymbol{Q}-\boldsymbol{QS\Delta})^n\boldsymbol{e}_1 \\ &= \left\{\sum_{m=0}^{n} {}_nC_m(\boldsymbol{I}-\boldsymbol{Q})^{n-m}(-\boldsymbol{QS\Delta})^m\right\}\boldsymbol{e}_1 \\ &= \left\{(\boldsymbol{I}-\boldsymbol{Q})^n + n(\boldsymbol{I}-\boldsymbol{Q})^{n-1}(-\boldsymbol{QS\Delta})\right. \\ &\quad \left. + \sum_{m=2}^{n} {}_nC_m(\boldsymbol{I}-\boldsymbol{Q})^{n-m}(-\boldsymbol{QS\Delta})^m\right\}\boldsymbol{e}_1. \quad (18)\end{aligned}$$

Here, ${}_nC_m$ is the binomial coefficient.

Now, we define $\tilde{Q}_n$ as

$$\tilde{Q}_n = \sum_{m=1}^{n} {}_nC_m Q^m(-1)^{m+1} \quad (19)$$

and replace Q with $\tilde{Q}_n$. Then, the convergence equation changes to

$$\tilde{\boldsymbol{e}}_{j+1,\,n} = \left(\boldsymbol{I}-\tilde{\boldsymbol{Q}}_n - \tilde{\boldsymbol{Q}}_n\boldsymbol{S\Delta}\right)^j \boldsymbol{e}_1 \quad (20)$$

where $\tilde{\boldsymbol{e}}_{j,\,n}$ denotes the error vector of the jth iteration when $\tilde{Q}_n$ is used. Then, the monotonic convergence condition is given as

$$\|\boldsymbol{I}-\tilde{\boldsymbol{Q}}_n - \tilde{\boldsymbol{Q}}_n\boldsymbol{S\Delta}\|_\infty < 1. \quad (21)$$

Here, the tracking error vector of the second iteration $\tilde{\boldsymbol{e}}_{2,\,n}$ is written as follows.

- When $n = 2$ $(\tilde{Q}_2 = 2Q - Q^2)$

$$\begin{aligned}\tilde{\boldsymbol{e}}_{2,\,2} &= \left\{\boldsymbol{I} - (2\boldsymbol{Q}-\boldsymbol{Q}^2) - (2\boldsymbol{Q}-\boldsymbol{Q}^2)\boldsymbol{S\Delta}\right\}\boldsymbol{e}_1 \\ &= \left\{(\boldsymbol{I}-\boldsymbol{Q})^2 - 2(\boldsymbol{I}-\boldsymbol{Q})\boldsymbol{QS\Delta} - \boldsymbol{Q}^2\boldsymbol{S\Delta}\right\}\boldsymbol{e}_1. \quad (22)\end{aligned}$$

- When $n = 3$ $(\tilde{Q}_3 = 3Q - 3Q^2 + Q^3)$

$$\begin{aligned}\tilde{\boldsymbol{e}}_{2,\,3} &= \left\{\boldsymbol{I} - (3\boldsymbol{Q} - 3\boldsymbol{Q}^2 + \boldsymbol{Q}^3) - (3\boldsymbol{Q} - 3\boldsymbol{Q}^2 + \boldsymbol{Q}^3)\boldsymbol{S\Delta}\right\}\boldsymbol{e}_1 \\ &= \left\{(\boldsymbol{I}-\boldsymbol{Q})^3 - 3(\boldsymbol{I}-\boldsymbol{Q})^2\boldsymbol{QS\Delta}\right. \\ &\quad \left. - 3(\boldsymbol{I}-\boldsymbol{Q})\boldsymbol{Q}^2\boldsymbol{S\Delta} - \boldsymbol{Q}^3\boldsymbol{S\Delta}\right\}\boldsymbol{e}_1. \quad (23)\end{aligned}$$

- When n $(\tilde{Q}_n = \sum_{m=1}^{n} {}_nC_m Q^m(-1)^{m+1})$

$$\begin{aligned}\boldsymbol{e}_{2,\,n} &= \left[\boldsymbol{I} - \sum_{m=1}^{n} {}_nC_m\boldsymbol{Q}^m(-1)^{m+1}\right. \\ &\quad \left. - \left\{\sum_{m=1}^{n} {}_nC_m\boldsymbol{Q}^m(-1)^{m+1}\right\}\boldsymbol{S\Delta}\right]\boldsymbol{e}_1 \\ &= \left[(\boldsymbol{I}-\boldsymbol{Q})^n + n(\boldsymbol{I}-\boldsymbol{Q})^{n-1}(-\boldsymbol{QS\Delta})\right. \\ &\quad \left. - \left\{\sum_{m=2}^{n} {}_nC_m(\boldsymbol{I}-\boldsymbol{Q})^{n-m}\boldsymbol{Q}^m\right\}\boldsymbol{S\Delta}\right]\boldsymbol{e}_1. \quad (24)\end{aligned}$$

The proof of (24) is described in the Appendix. Compared with (18) and (24), the first and second terms on the right-hand side are identical. To guarantee the error convergence stability [see (15)], Q is generally designed with a small gain in the high-frequency band where $|S\Delta|$ has a high gain, as shown in Fig. 14. Then, the third terms of the right-hand sides of (18) and (24) are well attenuated by Q^m. Therefore, if the effect of the third term with Q^m is much smaller than that of the second term with Q, $\tilde{\boldsymbol{e}}_{2,\,n}$ is written as

$$\tilde{\boldsymbol{e}}_{2,\,n} \approx \boldsymbol{e}_{n+1}. \quad (25)$$

Equation (25) indicates that the error that appears after 1-time learning using $\tilde{Q}_n$ is almost equal to the error that appears after n-times learning using Q.

B. Convergence With No Modeling Error

If there is no modeling error, i.e., $\boldsymbol{\Delta} = \boldsymbol{O}$, (25) becomes completely equal as in the following:

$$\boldsymbol{e}_{n+1} = \tilde{\boldsymbol{e}}_{2,\,n}. \quad (26)$$

From (18) and (24), the following equation is obtained:

$$\boldsymbol{e}_{jn+1} = (\boldsymbol{I}-\boldsymbol{Q})^{jn}\boldsymbol{e}_1 = \left\{(\boldsymbol{I}-\boldsymbol{Q})^n\right\}^j\boldsymbol{e}_1 = \tilde{\boldsymbol{e}}_{j+1,\,n}. \quad (27)$$

Fig. 1 shows an intuitive understanding of this equation. Therefore, the conventional Q filter requires jn iterations of learning to converge to $\boldsymbol{e}_{jn+1}$. In contrast, the n-times learning filter $\tilde{Q}_n$ requires only j iterations of learning to achieve the same $\boldsymbol{e}_{jn+1}$. Then, the proposed RPTC can converge the error n times faster than the conventional RPTC. In other words, when using the proposed high-order filter $\tilde{Q}_n$, the convergence of $\tilde{\boldsymbol{e}}_{j+1,\,n}$ can be estimated from the error $\boldsymbol{e}_{jn+1}$ of the original Q filter.

C. Realization of n-Times Learning Filter $\tilde{Q}_n$

From (19), n-times learning filter $\tilde{Q}_n$ is a multistage Q filter. Therefore, $\tilde{Q}_n$ is expressed as a finite-impulse response filter, as shown in the following:

$$\tilde{Q}_n = \sum_{m=-nN_q}^{nN_q} a_m z^m$$

where N_q is an integer that is defined in (2). This nonproper filter is realized as

$$\tilde{Q}_{nr} = z^{-nN_q}\tilde{Q}_n \quad (28)$$

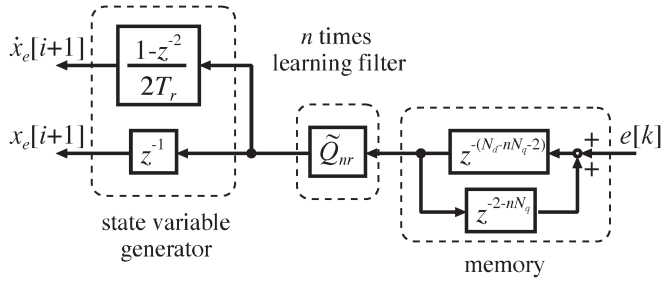

Fig. 10. PSG with the n-times learning filter.

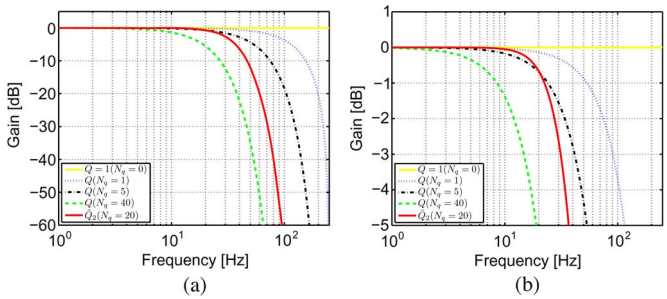

Fig. 11. Gain characteristics of Q and $\tilde{Q}_n$. (a) Properties of Q and $\tilde{Q}_n$. (b) Zoomed image of Q and $\tilde{Q}_n$.

where the suffix r implies "realization." The numbers of sample delays in (3) and (28) are different. Accordingly, we should change the number of sample delays in the memory to realize $\tilde{Q}_n$ similar to Fig. 10. Here, n is restricted as follows:

$$nN_q + 2 < N_d$$
$$n < \frac{N_d - 2}{N_q}. \tag{29}$$

D. Characteristic of n-Times Learning Filter $\tilde{Q}_n$

In this section, the characteristics of $\tilde{Q}_n$ are revealed. From (13), $\boldsymbol{e}_2$ becomes $\mathbf{0}$ after the first iteration when $\boldsymbol{\Delta} = \boldsymbol{O}$, and $\boldsymbol{Q} = \boldsymbol{I}$. In other words, the fastest convergence is achieved when $\boldsymbol{Q} = \boldsymbol{I}$. The gain characteristic of $\boldsymbol{Q} = \boldsymbol{I}$ is 0 dB for all frequencies. However, we should design Q as a low-pass filter because the control system becomes unstable due to the high-frequency signals that originate from noise or mechanical resonance. Therefore, to realize stable and fast convergence, filter Q should have a wide 0-dB region at low frequencies and a sharp rolloff property at high frequencies.

Fig. 11 shows the gain characteristics of Q and $\tilde{Q}_n$. Here, the sampling time is 2 ms. Note that the gain characteristics of Q_r and $\tilde{Q}_{nr}$ are the same with Fig. 11. The phase characteristic is zero at all frequencies because Q and $\tilde{Q}_n$ are zero-phase low-pass filters. Fig. 11(b) shows the plots of $Q(N_q = 0,\ 1,\ 5,\ 40)$ and $\tilde{Q}_2(N_q = 20)$. As N_q increases, the high-frequency gain of Q decreases, and high robust stability of convergence is obtained. However, the frequency bandwidth through which 0-dB signals can pass decreases with bigger N_q, which results in slower convergence and worse error attenuation. In contrast, the proposed $\tilde{Q}_2(N_q = 20)$ can yield a higher bandwidth and a sharper rolloff property than the conventional $Q(N_q = 5)$, as shown in Fig. 11(b).

In Fig. 12, the variation in the gain characteristic of $\tilde{Q}_n$ is discussed using various parameters of n and N_q. As shown in Fig. 12(a), a sharper rolloff characteristic is obtained as n increases with fixed $N_q = 20$. In Fig. 12(b), the filter's order is fixed to $nN_q = 40$. In comparison with $\tilde{Q}_2(N_q = 20)$, the bandwidth of the conventional $Q(N_q = 40)$ and $\tilde{Q}_4(N_q = 10)$ decreases and increases, respectively. In this comparison, it seems that N_q is more dominant than n because the bandwidth considerably changes and the rolloff characteristics almost remain unchanged.

As a result, it can be said that the proposed method can afford a greater number of degrees of freedom for tuning the bandwidth and the rolloff characteristics. For example, it is possible that the proposed $\tilde{Q}_n$ of $n = 5$ and $N_q = 20$ has a slightly higher bandwidth and a considerably larger rolloff than the conventional $Q(N_q = 1)$, as shown in Fig. 12(c). Generally, in the learning control, there is a tradeoff between the convergence speed and the robustness of stability. $\tilde{Q}_5(N_q = 20)$ is a more suitable learning filter than $Q(N_q = 1)$ when a sharper rolloff property and a high bandwidth are needed.

The rolloff characteristic of the conventional learning filter Q is improved by sacrificing the frequency bandwidth. In contrast, both the rolloff and bandwidth of the proposed n-times learning filter $\tilde{Q}_n$ can be improved by increasing n and N_q, respectively.

V. Simulation

The effectiveness of the n-times learning RPTC is verified by performing two simulations.

A. Simulation of Ideal Condition With Sinusoidal Disturbance

n-Times fast convergence is confirmed by the simulation of the nominal plant model $P_n(s)$ with sinusoidal disturbance. The position error due to this disturbance is suppressed with the RPTC.

A multirate FF controller is designed for P_n. Then, this simulation condition is ideal because the modeling error is zero. A PID FB controller with a position frequency bandwidth of 40 Hz is designed following the pole placement method. Moreover, Q and 3-times learning filter $\tilde{Q}_3$ are designed with $N_q = 20$ and T_r $(= 2T_u$, $T_u = 1$ ms, where T_u is the control period [16]). The input disturbance that is shown in Fig. 13(a) is suppressed by the conventional RPTC and the proposed 3-times learning RPTC. For the sake of clear comparison, iteration cycle T_d is considered twice the disturbance cycle $(T_d = 0.5\ \mathrm{s})$.

The simulation results are shown in Fig. 13. Fig. 13(b) shows the tracking error. Moreover, Fig. 13(c) and (d) are zoomed-in versions of parts of Fig. 13(b). The tracking error is suppressed after T_d owing to RPTC compensation. Very small impulse-shaped tracking errors, as shown in Fig. 13(c), are generated by the discontinuous state variables of the multirate FF controller. In Fig. 13(d), the conventional fourth-cycle error $\boldsymbol{e}_4$ is equal to the second-cycle error $\tilde{\boldsymbol{e}}_{2,3}$ of the proposed RPTC.

In contrast, the second-cycle error $\tilde{\boldsymbol{e}}_{2,3}$ is only achieved by the 1-time learning with the proposed RPTC. Therefore, the tracking error of the proposed RPTC can converge three times faster than the conventional RPTC. In addition, the impulse-shaped errors converged faster using the proposed RPTC. Therefore, the convergence analysis in Section IV-B is verified.

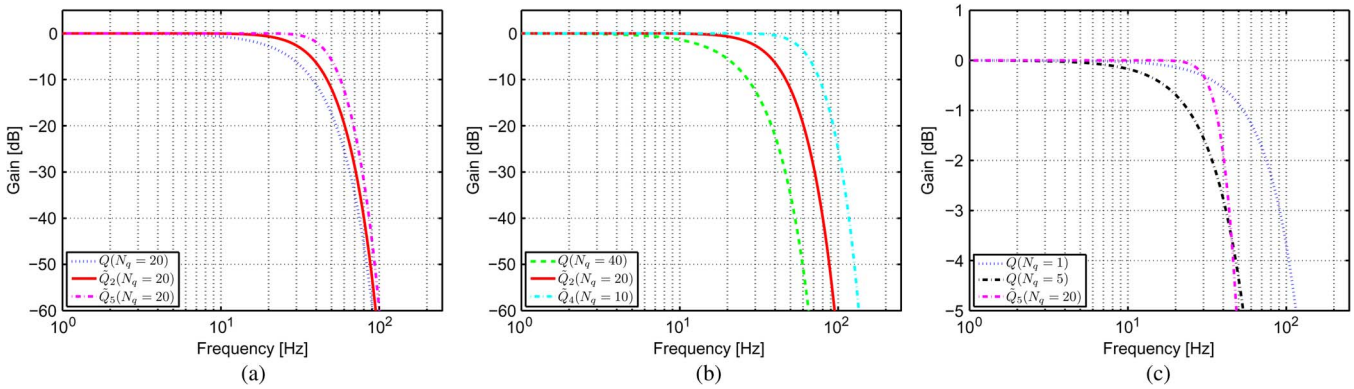

Fig. 12. Gain tuning method for $\tilde{Q}_n$. (a) Q and $\tilde{Q}_n (n = 2, 5)$ with fixed $N_q = 20$. (b) Q and $\tilde{Q}_n$ with fixed order $nN_q = 40$. (c) Zoomed image of $Q(N_q = 1, 5)$ and $\tilde{Q}_5(N_q = 20)$.

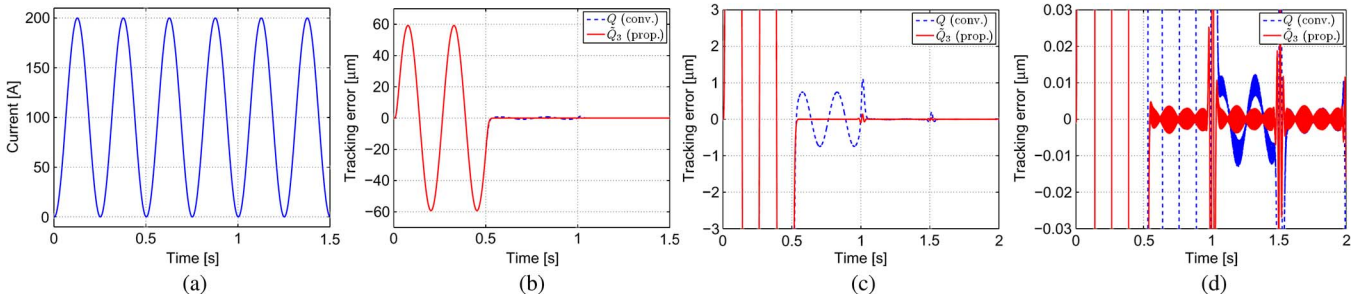

Fig. 13. Simulation results (ideal condition). (a) Input disturbance. (b) Error. (c) Error (zoomed image). (d) Error (further zoomed image).

TABLE I
PARAMETERS OF THE RESONANCE MODES

i	K_i	ω_i rad/s	ζ_i
1	60	2π 50	1
2	40	2π 200	0.3
3	65	2π 565	0.04

B. Simulation With Detailed Model of Ball-Screw-Driven Stage

The effectiveness of the proposed method was also verified by performing a detailed simulation of the ball-screw-driven stage. The nominal plant for designing the PTC is a rigid-body model $P_n(s)$ [see (1)]. Simulation model $P(s)$ includes three resonance modes with deadtime $T_d = 70$ μs as the modeling error as in the following:

$$P(s) = \left(\frac{K_T}{Js^2 + Ds} + \sum_{i=1}^{3} \frac{K_i}{s^2 + 2\zeta_i \omega_i s + \omega_i^2} \right) e^{-T_d s}. \quad (30)$$

The parameters are listed in Table I. Furthermore, nonlinear rolling friction is injected as the input disturbance, which is obtained from the experiments on a data-based friction model [14]. Owing to this nonlinear friction, a periodic tracking error is generated when a periodic position reference is given. In this case, the RPTC suppresses the periodic tracking error through learning. FB controller $C_{\rm fb}(s)$ is designed following the pole placement method with a position-loop bandwidth of 45 Hz and is discretized by the Tustin transformation with $T_u = 1$ ms.

Fig. 14 shows the frequency response of multiplicative modeling error $\Delta(s) = (P(s) - P_n(s))/P_n(s)$ and sensitivity function $S_n(s) = (1 + P_n(s)C_{fb}(s))^{-1}$. Because $|\Delta(s)S_n(s)|$ has a high gain in a higher frequency band, $\tilde{Q}_5$ is designed as the proposed learning filter with $n = 5$ and $N_q = 20$ to guarantee the error convergence. Moreover, the conventional filter Q is designed with $N_q = 20$.

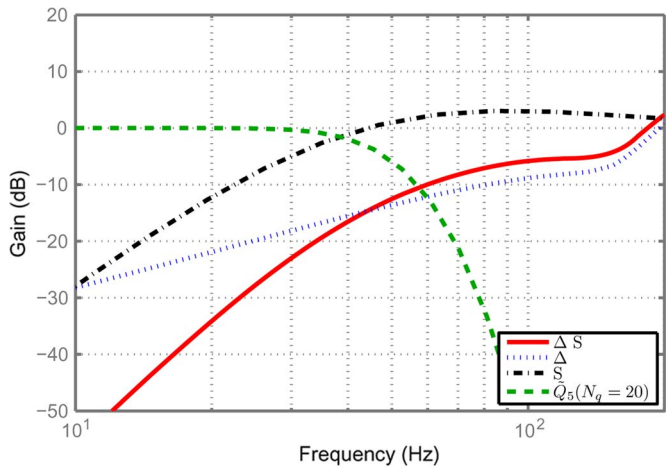

Fig. 14. Q and $\tilde{Q}_n (n = 2, 5)$ with fixed $N_q = 20$.

The simulation results are shown in Fig. 15. The sinusoidal position reference that is shown in Fig. 15(a) is given to the stage. Then, the tracking error occurs, as shown in Fig. 15(b). Fig. 15(c) is a close-up of the third iteration error. From Fig. 15(b) and (c), the maximum tracking error of each cycle decreases with the 5-times learning RPTC. Therefore, $\tilde{Q}_5$ can learn the tracking error more effectively than the conventional Q.

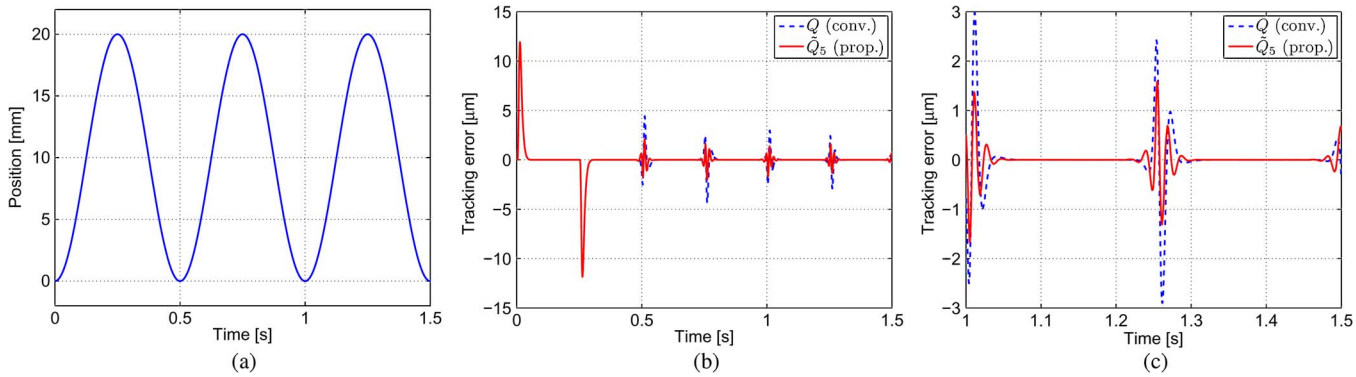

Fig. 15. Simulation results (with nonlinear friction). (a) Target trajectory. (b) Error. (c) Error (zoomed image of the third iteration).

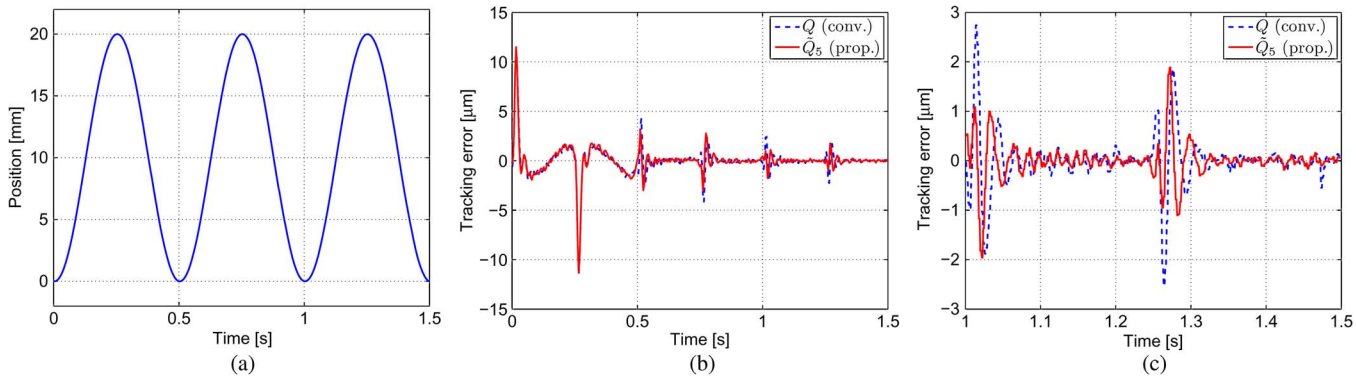

Fig. 16. Experimental results of the n-times learning control. (a) Target trajectory. (b) Error. (c) Error (zoomed image of the third iteration).

Five-times fast convergence is not exactly achieved because the disturbance that is caused by the nonlinear friction is not perfectly periodic. However, it can be confirmed that quick convergence is achieved by the proposed n-times learning, even with modeling error.

In these simulations, the modeling errors due to mechanical resonance modes are mainly considered. In contrast, the RPTC method is robust against parameter variations of the rigid-body mode, such as K_T, J, and T_d. The error convergence performance was analyzed under the parameter variation in [13].

VI. Experiment

A. Verification of n-Times Learning Control

The proposed method was implemented with the experimental ball-screw-driven system. Controllers and learning filters were the same as in the detailed simulation of the ball-screw-driven stage. A multirate FF controller and a PID controller are designed identical to those in the simulation. Q and $\tilde{Q}_5$ are designed with a sampling time of 2 ms, and $N_q = 20$.

The experimental results are shown in Fig. 16. Fig. 16(a) is the target trajectory, which is the same as in the simulation. Fig. 16(b) and (c) are plots of the tracking error and the zoom of the third-cycle tracking error, respectively. In Fig. 16(b) and (c), the maximum tracking error of the proposed RPTC is smaller than that of the conventional RPTC, which is similar with the simulation. Hence, fast convergence was verified from the simulation and experimental results.

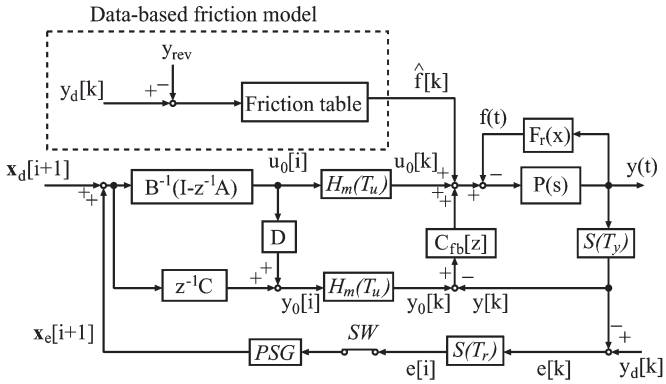

Fig. 17. Combination of the friction compensation with the RPTC.

B. Combination With Model-Based Compensation

Although the learning control is robust against plant and disturbance variations, it is not effective for the first iteration of learning. Then, in the case of the NC machine tool application, a quadrant glitch is not avoidable with only the learning control during the learning process. Therefore, the proposed learning control is combined with the rolling friction compensation.

Both the conventional RPTC and the proposed RPTC can be easily combined with model-based compensation methods, i.e., with both the equation-based and data-based friction models. In this section, the data-based method [14] is selected as an example. The block diagram of the proposed control system is shown in Fig. 17. Each $y_{\rm rev}$ and $y_d[k]$ represents the position

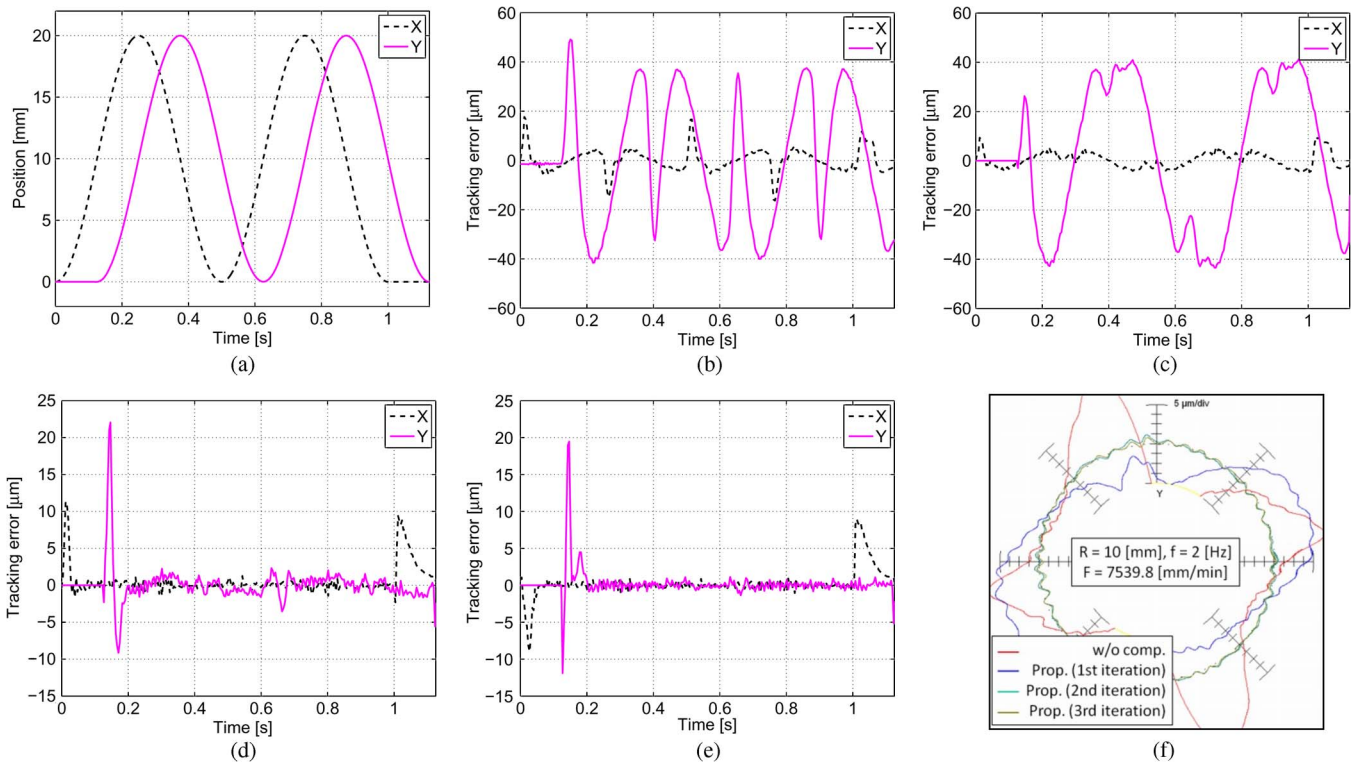

Fig. 18. Experimental results of the combination method. (a) Target trajectory. (b) Without compensation ($\hat{f} = 0$, No RPTC). (c) Combination method (first iteration, No RPTC). (d) Combination method (second iteration, RPTC). (e) Combination method (third iteration, RPTC). (f) Circularity measurement.

of velocity reversal and the target trajectory, respectively. The dotted line in this figure shows the data-based rolling friction model, and the input of the friction table is given by the difference between $y_d[k]$ and $y_{\rm rev}$. Therefore, the rolling friction model performs FF compensation $\hat{f}[k]$ to reject nonlinear friction $f(t)$ before the RPTC compensation starts. In addition, the remaining tracking error due to the friction model mismatch, the plant variation, or the modeling error can be suppressed by the RPTC in subsequent iterations.

Fig. 18 shows the experimental results of the combination method both for the x-axis and the y-axis. As shown in Fig. 18(b), 2-Hz sinusoidal target trajectories are given with an amplitude of 10 mm. This target is the severe condition for this experimental stage because its tangential speed is $F = 7540$ mm/min, although commercial products with this stage normally use below $F = 1000$ mm/min. Fig. 18(b) shows the error in the case without friction compensation ($\hat{f}[k] = 0$) and without the use of the RPTC, in which the switch in Fig. 17 turns off. Because the plant parameters, such as inertia J and torque constant K_T, are not well identified, the y-axis has a large position error.

In the first iteration, the RPTC does not work because of the learning process. Fig. 18(c) shows that the friction compensation can reject the quadrant glitch. Moreover, the remaining error that is caused by the modeling error, the parameter mismatch, and the x–y cross-coupling disturbance is reduced by the second and third iterations of the RPTC, as shown in Fig. 18(d) and (e), because the frequency of the trajectories of 2 Hz is less than the bandwidth of the learning filter. Note that, at $t = 0.125$ s of the y-axis and at $t = 0$ and 1 s of the x-axis, a relatively large error occurs because of the discontinuous waveform of the target velocity trajectory. To perfectly track this trajectory, a very large control input is required. Although this can be improved by inputting a smoother trajectory, we will not discuss it in this paper because it is not the main issue.

Fig. 18(f) shows the experimental results of the circularity measurement test that is detected by the grid encoder that is shown in Fig. 2. Only the data of the second period ($t = 0.5 \sim 1$) is plotted. Using the proposed combined method with the iteration, the roundness error converges to less than 3 μm, which is much better than the result of the commercial product with the same stage in this severe condition.

VII. Conclusion

In this paper, we have proposed an n-times learning RPTC that can converge the tracking error n times faster than the conventional RPTC. The n-times learning filter that is used in the proposed RPTC has better low-pass and sharp rolloff characteristics than the conventional learning filter. n-Times fast convergence is verified from the results of an ideal simulation. The effectiveness of the proposed method is shown by the simulation and experiment with the x-axis of the ball-screw-driven stage of an NC machine tool. Finally, the proposed method is combined with the model-based friction compensation. The advantage of the method is experimentally confirmed by the x- and y-axes of the same stage, and the circularity is precisely evaluated.

APPENDIX
PROOF OF (24)

Matrix $\{(\boldsymbol{I}-\boldsymbol{Q})+\boldsymbol{Q}\}^n$ is expanded using the binomial theorem as follows:

$$\begin{aligned}\{(\boldsymbol{I}-\boldsymbol{Q})+\boldsymbol{Q}\}^n &= \sum_{m=0}^{n} {}_nC_m(\boldsymbol{I}-\boldsymbol{Q})^{n-m}\boldsymbol{Q}^m \\ \boldsymbol{I} &= (\boldsymbol{I}-\boldsymbol{Q})^n+n(\boldsymbol{I}-\boldsymbol{Q})^{n-1}\boldsymbol{Q} \\ &\quad + \sum_{m=2}^{n} {}_nC_m(\boldsymbol{I}-\boldsymbol{Q})^{n-m}\boldsymbol{Q}^m \\ \boldsymbol{I} &= \boldsymbol{I}+\sum_{m=1}^{n} {}_nC_m\boldsymbol{Q}^m(-1)^m+n(\boldsymbol{I}-\boldsymbol{Q})^{n-1}\boldsymbol{Q} \\ &\quad + \sum_{m=2}^{n} {}_nC_m(\boldsymbol{I}-\boldsymbol{Q})^{n-m}\boldsymbol{Q}^m. \end{aligned} \tag{31}$$

Therefore, we obtain

$$\begin{aligned}-\sum_{m=1}^{n} {}_nC_m\boldsymbol{Q}^m(-1)^{m+1} &= n(\boldsymbol{I}-\boldsymbol{Q})^{n-1}(-\boldsymbol{Q}) \\ &\quad - \sum_{m=2}^{n} {}_nC_m \\ &\quad \times (\boldsymbol{I}-\boldsymbol{Q})^{n-m}\boldsymbol{Q}^m. \end{aligned} \tag{32}$$

Moreover, from the binomial theorem, it is derived that

$$(\boldsymbol{I}-\boldsymbol{Q})^n = \boldsymbol{I}+\sum_{m=1}^{n} {}_nC_m\boldsymbol{Q}^m(-1)^m. \tag{33}$$

By multiplying $\boldsymbol{S\Delta}$ to (32) and by adding (33), (24) is proven.

REFERENCES

[1] L. Mostefai, M. Denai, O. Sehoon, and Y. Hori, "Optimal control design for robust fuzzy friction compensation in a robot joint," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 3832–3839, Oct. 2009.

[2] J. Na, Q. Chen, X. Ren, and Y. Guo, "Adaptive prescribed performance motion control of Servo mechanisms with friction compensation," *IEEE Trans. Ind. Electron.*, vol. 61, no. 1, pp. 486–494, Jan. 2014.

[3] Z. Jamaludin, H. Van Brussel, and J. Swevers, "Friction compensation of an XY feed table using friction-model-based feedforward and an inverse-model-based disturbance observer," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 3848–3853, Oct. 2009.

[4] H. Asaumi and H. Fujimoto, "Proposal on nonlinear friction compensation based on variable natural length spring model," in *Proc. SICE Annu. Conf.*, 2008, pp. 2393–2398.

[5] Y. Maeda and M. Iwasaki, "Rolling friction model-based analyses and compensation for slow settling response in precise positioning," *IEEE Trans. Ind. Electron.*, vol. 60, no. 12, pp. 5841–5853, Dec. 2013.

[6] Y. Maeda and M. Iwasaki, "Mode switching feedback compensation considering rolling friction characteristics for fast and precise positioning," *IEEE Trans. Ind. Electron.*, vol. 61, no. 2, pp. 1123–1132, Feb. 2014.

[7] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Syst. Mag.*, vol. 26, no. 3, pp. 96–114, Jun. 2006.

[8] D. Huang, J. X. Xu, V. Venkataramanan, and T. C. T. Huynh, "High-performance tracking of piezoelectric positioning stage using current-cycle iterative learning control with gain scheduling," *IEEE Trans. Ind. Electron.*, vol. 61, no. 2, pp. 1085–1098, Feb. 2014.

[9] K. Seki, M. Iwasaki, M. Kawafuku, H. Hirai, and K. Kishida, "Practical controller design of hybrid experimental system for seismic tests," *IEEE Trans. Ind. Electron.*, vol. 56, no. 3, pp. 628–634, Mar. 2009.

[10] H. Fujimoto, "Robust repetitive perfect tracking control of HDDs based on re-learning scheme," *IEEJ J. Ind. Appl.*, vol. 2, no. 1, pp. 40–47, 2013.

[11] K. K. Chew and M. Tomizuka, "Digital control of repetitive errors in disk drive systems," *IEEE Control Syst. Mag.*, vol. 10, no. 1, pp. 16–20, Jan. 1990.

[12] S. Hara, Y. Yamamoto, T. Omata, and M. Nakano, "Repetitive control system–A new-type servo system," *IEEE Trans. Autom. Control*, vol. 33, no. 7, pp. 659–668, Jul. 1988.

[13] H. Fujimoto, "RRO compensation of hard disk drives with multirate repetitive perfect tracking control," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 3825–3831, Oct. 2009.

[14] T. Takemura and H. Fujimoto, "Proposal of novel rolling friction compensation with data-based friction model for ball screw driven stage," in *Proc. 36th Annu. Conf. IEEE Ind. Electron. Soc.*, 2010, pp. 1932–1937.

[15] T. Takemura and H. Fujimoto, "High precision control of ball screw driven stage using repetitive control with sharp roll-off learning filter," in *Proc. 12th IEEE Int. Workshop Adv. Motion Control*, 2012, pp. 1–6.

[16] H. Fujimoto, Y. Hori, and A. Kawamura, "Perfect tracking control based on multirate feedforward control with generalized sampling periods," *IEEE Trans. Ind. Electron.*, vol. 48, no. 3, pp. 636–644, Jun. 2001.

[17] K. L. Moore, *Iterative Learning Control for Deterministic Systems*. New York, NY, USA: Springer-Verlag, 1993.

Hiroshi Fujimoto (S'99–M'01–SM'12) received the Ph.D. degree from The University of Tokyo, Tokyo, Japan, in 2001.

In 2001, he joined the Department of Electrical Engineering, School of Engineering, Nagaoka University of Technology, Nagaoka, Japan, as a Research Associate. From 2002 to 2003, he was a Visiting Scholar with the School of Mechanical Engineering, College of Engineering, Purdue University, West Lafayette, IN, USA. In 2004, he joined the Department of Electrical and Computer Engineering, Graduate School of Engineering, Yokohama National University, Yokohama, Japan, as a Lecturer, and in 2005, he went on to become an Associate Professor. Since 2010, he has been an Associate Professor with the Department of Advanced Energy, Division of Transdisciplinary Studies, Graduate School of Frontier Sciences, The University of Tokyo, Kashiwa, Japan. His research interests include control engineering, motion control, nanoscale servo systems, electric vehicle control, and motor drives.

Dr. Fujimoto is a member of the Society of Instrument and Control Engineers, the Robotics Society of Japan, and the Society of Automotive Engineers of Japan, and he is a senior member of the Institute of Electrical Engineers of Japan. He received the Best Paper Award from the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS in 2001, the Isao Takahashi Power Electronics Award in 2010, and the Best Author Prize of the Society of Instrument and Control Engineers in 2010.

Tadashi Takemura (S'10) received the B.S. and M.S. degrees in electrical and computer engineering from Yokohama National University, Yokohama, Japan, in 2010 and 2012, respectively.

From 2010 to 2012, he was a Research Student with The University of Tokyo, Kashiwa, Japan. He is currently with IHI Corporation, Tokyo, Japan.