Combined Feedforward Control and Disturbance Rejection Control Design for a Wafer Stage: A Data-Driven Approach Based on Iterative Parameter Tuning

MINGSHENG CAO1,2, YUMENG BO1,2, AND HUIBIN GAO1
1Changchun Institute of Optics, Fine Mechanics, and Physics, Chinese Academy of Sciences, Changchun 130033, China
2University of Chinese Academy of Sciences, Beijing 100049, China
Corresponding author: Mingsheng Cao (caomingsheng16@mails.ucas.edu.cn)

ABSTRACT This article presents a data-driven algorithm that combines the advantages of iterative feedforward tuning and disturbance rejection control to satisfy the precision requirements and ensure extrapolation capability of wafer scanning. The proposed algorithm differs from pre-existing algorithms in terms of its low requirement of system model, high extrapolation capability for non-repetitive trajectory tracking tasks, and high tracking precision. The feedforward controller is tuned based on instrumental variables. It utilizes tracking errors from past iterations to eliminate reference-induced errors without requiring a system model. Meanwhile, the system inverse is approximated during iterative process, and then a disturbance rejection control based on iterative tuning is constructed to compensate for disturbance-induced errors. The proposed algorithm is applied to a wafer stage. The experimental results validate the effectiveness and superiority of the proposed algorithm.

INDEX TERMS Data-driven, disturbance rejection control, feedforward control, iterative parameter tuning, wafer stage.

I. INTRODUCTION

Gene sequencing is an important scientific technology. In the production steps of the next-generation gene sequencing platform, a two-dimensional spot-array is attached to a patterned wafer substrate, and each spot contains a DNA nanoball, which is an amplified DNA cluster. The nanoball is imaged using an optical system while it is moving in a constant-velocity scanning motion. The DNA nanoball is positioned using a wafer stage. The control performance of the wafer stage determines the imaging efficiency and quality of the gene sequencing. Therefore, the following requirements are imposed on the controller:

1) high tracking precision;
2) strong resistance to disturbance;
3) high extrapolation capability for non-repetitive trajectory tracking tasks.

Feedforward control is widely used in motion systems, in particular in systems requiring high precision in trajectory tracking tasks. Feedforward controllers can compensate for errors induced by known reference trajectories and repetitive disturbances; therefore, they can improve system performance significantly using a priori knowledge. Traditional feedforward control approaches, including iterative learning control (ILC), model-based feedforward control, and iterative feedforward tuning (IFFT), can satisfy these requirements.

Systems with a repetitive nature, such as a wafer stage, lithography machine, and mechanical arm, have repeating trajectories that are known a priori. Iterative learning control is a feedforward control method designed for repetitive tasks. It was proposed by Arimoto et al. [1] in 1984 and in the past three decades has been extended by researchers to different schemes, such as inverse systems [2] and the stochastic method [3], [4]. Furthermore, ILC can be combined with other control methods, including adaptive control [5], [6] and robust control [7], [8], to achieve better performances based on system characteristics and requirements.

The ILC algorithm can effectively reduce repetitive disturbance and achieve high control precision [9]. However, non-repetitive disturbance may accumulate and negatively affect the control performance.

The associate editor coordinating the review of this manuscript and approving it for publication was Jesus Felez.
affect the convergence of ILC [10], [11]. Furthermore, when extrapolating to different tasks, changes in the reference trajectory can result in performance deterioration. Poor extrapolation results in strict restrictions when applied to industrial plants.

By contrast, model-based feedforward control results in high control precision and high extrapolation capability for a class of tasks [12]. The model-based control method requires a system model and uses its inverse or a sensitivity function as a learning filter to compensate for system information. Hence, the control performance is highly dependent on the quality and accuracy of the system model obtained from system identification. Moreover, complicated dynamics and different unfavorable environmental conditions render it difficult to achieve an effective control [13]–[15].

The requirement of a high-quality model conflicts with the industrial expectation of a simple model-free and data-driven approach [16].

To combine the high control precision of ILC with the extrapolation capability of model-based feedforward control and remove the requirement of system models, many approaches have been investigated. In 2010, Wijdeven and Bosgra proposed a method that introduced basis functions [17]. With a basis function, a signal is projected onto a space spanned by the basis function, and a feedforward controller can be used to approximate the system inverse. Furthermore, the feedback controller can be parameterized, as shown in [18]. By introducing the basis function, the requirement of system model identification is transformed into an IFFT problem of a parameterized feedforward controller [19]. In [20], a parameterized feedforward controller was tuned using a data-driven gradient-descent learning algorithm. In [20]–[22], it was shown that using a least-squares algorithm resulted in bias errors. To eliminate those bias errors, a least-squares algorithm with an instrumental variable (IV) was proposed, which resulted in an unbiased parameter estimation. Accuracy analysis was provided in [23]. Song et al. [24] proposed a high-order approach based on an IV using error data from all past iterations to achieve a high tolerance for disturbance. In [25], [26], a rational basis function was introduced to achieve a higher precision.

A feedforward controller can compensate for errors induced by known reference trajectories rather than non-repetitive disturbance-induced errors. To eliminate non-repetitive disturbance-induced errors, several types of disturbance rejection controls have been investigated, such as the disturbance observer based (DOB) control [27], active disturbance rejection control (ADRC) [28], and extended state observer (ESO) [29]. Herein, a new data-driven iterative-tuning-based compound control strategy that combines both feedforward control and disturbance rejection control is proposed to improve the motion performance of the wafer stage. The proposed disturbance rejection controller aims to eliminate the disturbance-induced error, whereas the feedforward controller aims to compensate for the reference trajectory-induced error. Meanwhile, when the reference trajectory changes, the algorithm still maintains an acceptable precision. Additionally, the proposed algorithm is data-driven and does not require a priori information of the system; therefore, the algorithm is applicable to different plants without requiring system identification in advance.

Compared to previous studies, the main contributions of this study are as follows:

1) A data-driven algorithm that combines feedforward control and disturbance rejection control is proposed, thereby satisfying control precision and extrapolation capability simultaneously without requiring a system model or sensitivity function.

2) Both feedforward control and disturbance rejection control can be constructed via iterative tuning. The design approach is simple and easy to implement.

3) Experiments are performed on a wafer stage with varying reference trajectories; subsequently, they are compared with existing ILC and IFFT methods. The validity of the proposed algorithm is proved.

The paper is organized as follows. Section II describes the control configuration of the system and the control problem considered herein. In Section III, a detailed statement of the components used and the procedure of the proposed algorithm are provided. The experimental setup employed to assess the performance of the proposed algorithm and the experimental results are provided in Section IV. Conclusions are presented in Section V.

II. PROBLEM FORMULATION

A. CONTROL CONFIGURATION

Let the superscript \( k \) represent the trial number. Consider the control configuration as depicted in Fig. 1, where \( P \) is a true unknown system that is assumed to be single-input-single-output and linear time-invariant. \( C_{fb} \) is a feedback controller designed to stabilize system \( P \) by introducing a feedback control signal. \( C_{ff}^k \) represents the feedforward controller that is part of the disturbance observer. Furthermore, the known reference trajectory \( r \) is designed as a fourth-order polynomial trajectory with constraints on the first three derivatives. \( y^k \) denotes the output signal of the system, \( u^k \) the feedforward control signal produced by \( C_{ff}^k \), and \( d^k \) the output signal of the controller, \( d^k \) the unknown disturbance whose mean is assumed to be zero, and \( \hat{d}^k \) the estimate of the unknown disturbance. All signals considered are discrete-time signals.

Based on Fig.1, the measured signal \( y^k \) is expressed as \( y^k = y_r^k + y_d^k \), with

\[
y_r^k = SP(C_{ff}^k + C_{fb})r, \tag{1}
\]

\[
y_d^k = SP(d^k - \hat{d}^k), \tag{2}
\]

where \( S = (I + PC_{fb})^{-1} \) denotes the sensitivity function of the system, and

\[
\hat{d}^k = \frac{PC_{ff}^k - 1}{PC_{ff}^k}(C_{ff}^k r + C_{fb}e^k) + d^k. \tag{3}
\]
The tracking error $e^k$ can be expressed as

$$e^k = r - y^k = Sr - SPC_{ff} r - SP(d^k - \hat{d}^k).$$  \hspace{1cm} (4)

This shows that the tracking error $e^k$ comprises two components: a reference trajectory-induced error

$$e^k_r = Sr - SPC_{ff} r$$  \hspace{1cm} (5)

and a disturbance-induced error

$$e^k_d = -SP(d^k - \hat{d}^k).$$  \hspace{1cm} (6)

From (3) and (4), it is clear that the estimate of unknown disturbance $\hat{d} = d$, and the tracking error is equal to 0 if

$$C_{ff}^k = P^{-1}. \hspace{1cm} (7)$$

In (7), feedforward controller $C_{ff}^k$ is introduced to approximate the inverted plant dynamics such that reference-induced errors can be compensated when passing through $C_{ff}$.

It is noteworthy that $P$, $S$, and $d^k$ are unknown; therefore, $e^k_r$ and $e^k_d$ cannot be constructed from the measured tracking error $e^k$. Hence, a data-driven iterative optimization problem is formulated in the next section to update $C_{ff}^k$. Namely, for a transfer function $G$ and a discrete signal $\mu$, the filtered signal of $\mu$ by $G$ can be expressed as $y = G\mu$.

**B. FEEDFORWARD CONTROLLER PARAMETERIZATION**

The traditional feedforward controller introduced to approximate the system inverse is designed from system information calculated a priori from system identification before initializing the control algorithm. Whenever an algorithm is applied to a new system, system identification must be conducted using system identification methods, such as frequency sweeping and correlation analysis. Consequently, this will increase the complexity of the control process.

To reduce the complexity of the control process and integrate the control process into a single algorithm, a parameterized feedforward controller $C_{ff}(\theta)$ is applied to acquire the feedforward control signal, thereby reducing trajectory-induced errors.

As shown in Fig.1, feedback control signal $u_{ff}^{k+1}$ is parameterized with respect to a reference trajectory $r$ and a fixed feedforward controller $C_{ff}^{k+1}(\theta^{k+1})$ as follows:

$$u_{ff}^{k+1} = C_{ff}^{k+1} r, \hspace{1cm} (8)$$

Subsequently,

$$C_{ff}^{k+1} = C_{ff}^k + C_{ff}^\Delta,$$  \hspace{1cm} (9)

where the update $C_{ff}^\Delta$ based on measured signals $e^{k+1}$ and $y^{k+1}$ is obtained from the optimization problem.

$$C_{ff}^\Delta = \arg \min_{C_{ff}^\Delta} V(C_{ff}^\Delta), \hspace{1cm} (10)$$

where $C$ is designed a priori as in Definition 1.

**Definition 1:** The feedforward controller $C_{ff}$ is parameterized in terms of the basis functions as follows:

$$C = \left\{ C_{ff}(\theta, q) \mid C_{ff}(\theta, q) = \sum_{i=1}^{n} \psi_{ff,i}(q)\theta_i \right\} \hspace{1cm} (11)$$

Here, $\theta \in \mathbb{R}^{n \times 1}$, the parameter vector of $C_{ff}$, is expressed as

$$\theta = [\theta_1, \theta_2, \ldots, \theta_n]^T, \hspace{1cm} (12)$$

and the polynomial basis functions are defined as

$$\psi_{ff}(q) = [\psi_1(q), \psi_2(q), \ldots, \psi_n(q)]. \hspace{1cm} (13)$$

All $\psi_i(q)$ are basis functions of $q$ as follows:

$$\psi_1(q) = \frac{1 - q^{-1}}{T_s},$$

$$\psi_2(q) = \left(\frac{1 - q^{-1}}{T_s}\right)^2,$$

$$\psi_n(q) = \left(\frac{1 - q^{-1}}{T_s}\right)^n. \hspace{1cm} (14)$$

where $T_s$ denotes the sampling time of the system, and $n$ represents the number of basis functions introduced into the parameterization. For simplicity, we will omit $q$ herein after.

The optimal parameter vector of the feedforward controller is a parameter vector satisfying the following definition.

**Definition 2:** The feedforward controller parameter vector $\theta^*$ is optimal if the feedforward controller designed with...
the corresponding parameter vector $C_{ff}^* = \Psi_{ff} \theta^*$ satisfies $e^* = S(1 - PC_{ff}^*)r = 0$.

The parameterized feedforward controller is designed to satisfy the following goal.

**Goal 1.** Determine $C_{ff}$ built based on $\theta$ such that the criterion $V(\theta^k)$ in (10) is minimized in the trajectory tracking process, where $V(\theta^k)$ is formulated as:

$$V(\theta^k) = \| e^k(\theta^k) \|_2^2,$$  

(15)

where $e^k$ is formulated as in (4).

**III. ALGORITHM**

In this section, a new algorithm that combines IV-based IFFT and disturbance rejection control (IFFT-DRC) is proposed to deal with the problems that arise during reference tracking. A theoretical formulation and a detailed procedure of the proposed algorithm are provided in the following.

**A. ITERATIVE PARAMETER TUNING APPROACH**

With respect to parametrization in Section II-B, measured signals $e^k$ and $y^k$ can be extended using parameter vector $\theta^k$ as follows:

$$e^k = S(1 - PC_{ff}^k(\theta^k))r - SP(d^k - \hat{d}^k),$$  

(16)

$$y^k = SP(C_{ff}^k(\theta^k) + C_{fb})r + y^*_d.$$  

(17)

From (17), we can express $r$ as:

$$SPr = (C_{ff}^k(\theta^k) + C_{fb})^{-1}(y^k - y^*_d).$$  

(18)

Evaluating the expected value of (18) yields

$$SP\mathbb{E}r = \mathbb{E}((C_{ff}^k(\theta^k) + C_{fb})^{-1}(y^k - y^*_d))$$  

$$= \mathbb{E}((C_{ff}^k(\theta^k) + C_{fb})^{-1}SPr).$$  

(19)

By combining (16) and (19), the gradient with respect to parameter vector $\theta$ is expressed as follows:

$$\frac{d e^k}{d \theta} = S(-P \frac{d C_{ff}^k(\theta)}{d \theta} r)$$  

$$= -SP\Psi_{ff} r$$  

$$= -SP\Psi_{ff}(C_{ff}^k(\theta^k) + C_{fb})^{-1}SPr^{-1}y^k$$  

$$= -\Psi_{ff}(C_{ff}^k(\theta^k) + C_{fb})^{-1}y^k.$$  

(20)

**Remark 1:** If $(C_{ff}^k(\theta^k) + C_{fb})^{-1}$ is unstable, then the stable inversion procedure in [21] can be employed to obtain the value of $(C_{ff}^k(\theta^k) + C_{fb})^{-1}$.

Using the gradient-descent optimal algorithm, we can iteratively calculate $\theta^{k+1}$ as follows:

$$\frac{d V}{d \theta} = 2 \frac{d e^T}{d \theta} e,$$  

(21)

$$\frac{d^2 V}{d \theta^2} = 2 \frac{d e^T}{d \theta} \frac{d e}{d \theta},$$  

(22)

$$\theta^{k+1} = \theta^k - \rho \frac{d^2 V}{d \theta^2}^{-1} \frac{d V}{d \theta},$$  

(23)

where $\frac{d e}{d \theta}$ is denoted by (20). By iteratively calculating $\theta^k$, the algorithm will converge to an optimal parameter vector $\theta^*$. Although gradient-descent IFFT is an effective method to calculate optimal parameter vector $\theta$ rapidly, it has been discovered that this method will be affected by noise or disturbances [20] and, therefore, will yield a biased approximation of $\theta$. In [21], an IV was introduced to eliminate the contribution of the noise signal and achieve an unbiased converged optimal $\theta$. More detailed information can be found in [19], [22]. In this study, IV Z was set as

$$Z = [\psi_1 r, \psi_2 r, \ldots, \psi_n r].$$  

(24)

By introducing IV $Z \in \mathbb{R}^{N \times n}$, the criterion in (10) is transformed to

$$V_2(\theta^k) = \| Z^T e^k(\theta^k) \|_2^2.$$  

(25)

**Remark 2:** $V(\theta^k)$ is changed to $V_2(\theta^k)$ to achieve an unbiased approximation of $\theta$. This will not change the main criterion of the entire control process.

Based on the criterion of (25) and $\frac{d e^k}{d \theta}$ calculated in (20), we can solve the optimal problem by setting the learning matrix $L \in \mathbb{R}^{n \times N}$ as

$$L = -\rho \left( \frac{1}{N} Z^T de^k \right)^{-1} \frac{1}{N} Z^T,$$  

(26)

where $\rho$ is the learning gain. Subsequently, we set the update law as

$$\theta^{k+1} = \theta^k + Le^k,$$  

(27)

to calculate the converged $\theta$ using an iterative tuning algorithm.

**Remark 3:** As shown in (20) and (24), IV Z and $\frac{d e^k}{d \theta}$ are correlated. Consequently, $Z^T de^k$ is nonsingular.

**Theorem 1:** For measured signals $e^k$ and $y^k$, consider the update law in (27), an unbiased estimate $\theta^*$ can be obtained as $k \to \infty$, i.e.,

$$E(\theta^k) \to \theta^*, k \to \infty.$$  

(28)

**Proof:** From (4), the tracking error is expressed as

$$e^k = S(1 - P\Psi_{ff} \theta^k)r + e^*_d.$$  

(29)

From Definition 2, $S = SPC_{ff}^* = S \Psi_{ff} \theta^*$; substituting it into (29) yields

$$e^k = SP\Psi_{ff}(\theta^* - \theta^k)r + e^*_d$$  

$$= SP\Psi_{ff} r(\theta^* - \theta^k) + e^*_d.$$  

(30)

Next, substituting (30) into (27) and using $LSP\Psi_{ff} r = -\rho$ yields

$$\theta^{k+1} = \theta^k + Le^k$$  

$$= \theta^k + LSP\Psi_{ff} r(\theta^* - \theta^k) + Le^*_d$$  

$$= (1 + \rho)\theta^k - \rho \theta^* + Le^*_d.$$  

(31)
When \( k \to \infty \),
\[
\theta^\infty = \theta^* - \frac{1}{\rho} L \dot{e}_d^k = \theta^* + \left( \frac{1}{N} Z^T \frac{d e_k}{d \theta^*} \right)^{-1} \frac{1}{N} Z^T e_d^k.
\] (32)

Because of the design of IV \( Z \) in (24), and as the known reference trajectory \( r \) and \( e_d \) are uncorrelated, we obtain
\[
E(\theta^\infty) = \theta^* + (Z^T \frac{d e_k}{d \theta^*})^{-1} E(Z^T e_d^k) = \theta^*.
\] (33)

This indicates that \( E(\theta^k) \) converges to \( \theta^* \) as \( k \to \infty \). This completes the proof.

**B. FEEDFORWARD AND DISTURBANCE REJECTION CONTROL BASED ON PARAMETER TUNING**

Based on (9) and (10), \( C_{ff}^{k+1} \) is expressed as
\[
C_{ff}^{k+1} = C_{ff}^k + C_{ff}^\Delta = \Psi_{ff} \theta^k + \Psi_{ff} L e^k
\]
\[
= \Psi_{ff} (\theta^k + L e^k) = \Psi_{ff} \theta^{k+1}.
\] (34)

Based on Fig.1, the estimate of disturbance is expressed as
\[
\hat{d}^k = Q(y_{C_{ff} - u^k}).
\] (35)

Subsequently, the controller is expressed as
\[
u^k = C_{ff}^k r + C_{ff} e^k - \hat{d}^k.
\] (36)

The role of the proposed feedforward controller is to eliminate the trajectory-induced error \( e_r^k \) rather than both \( e_r^k \) and the disturbance-induced error \( e_d^k \). Disturbance rejection control based on iterative parameter tuning solves this problem.

**Theorem 2:** For measured signals \( \dot{e} \), consider the control law (36), the following fact holds: \( \lim_{k \to \infty} \dot{e}^k(t) = 0 \).

**Proof:** As shown in Fig.1, when \( Q = 1 \), the disturbance can be estimated as
\[
\hat{d}^k = \frac{PC_{ff}^k - 1}{PC_{ff}^k} (C_{ff}^k r + C_{ff} e^k) + d^k,
\] (37)

From Theorem 1, we conclude that \( \theta^* \) is an optimal parameter of the feedforward controller. It follows that
\[
\dot{e}_r^k = S(1 - PC_{ff}^*) r = 0, \quad k \to \infty,
\] (38)

which indicates that \( C_{ff}^* = P^{-1} \). Substituting \( C_{ff}^* \) into (37) yields
\[
\dot{d}^k = \frac{PC_{ff}^* - 1}{PC_{ff}^*} (C_{ff}^* r + C_{ff} e^k) + d^k = d^k.
\] (39)

Hence, the disturbance-induced error
\[
\dot{e}_d^k = -SP(d^k - \hat{d}^k) = 0, \quad k \to \infty.
\] (40)

Consequently, the tracking error \( \dot{e}^k = 0, k \to \infty \). This completes the proof.

For the above calculations, the steps to solve optimal problem (11) can be summarized as follows:

**Procedure 1.** Estimation of \( \theta^k \) in the \( k^{th} \) iteration.

**Initialization.**

1. Determine the appropriate learning gain \( \rho, 0 < \rho < 1 \), to obtain a trade off between the convergence speed and the noise tolerance.
2. Determine the appropriate basis functions \( \theta \) to describe feedforward controller \( C_{ff} \) and reference trajectory \( r \).
3. Determine the appropriate initial filter coefficients \( \theta^0 \).
4. Set the initial values of \( e^k, \dot{d}^k \), and \( \dot{y}^k \) to 0.

**Main procedure.**

1. Perform an experiment using (36), and measure \( e^k \) and \( \dot{y}^k \). If the tracking error \( \dot{e}^k \) meets the accuracy requirement, terminate the iterations, else
2. Calculate \( \frac{d e_k}{d \theta^k} = \Psi_{ff} (C_{ff}^k + C_{ff}^\Delta)^{-1} \dot{e}_d^k \) using (13), (27), and (34).
3. Construct IV as \( Z = [\psi_1 r, \psi_2 r, \ldots, \psi_n r] \) using (14).
4. Calculate \( L = -\rho (\frac{1}{N} Z^T \frac{d e_k}{d \theta^k})^{-1} \frac{1}{N} Z^T \).
5. Calculate \( \theta^k+1 = \theta^k + L e^k \) and \( C_{ff}^{k+1} = \Psi_{ff} \theta^k+1 \).
6. Set \( k \to k + 1 \) and return to Step (1).

**Remark 4:** The traditional disturbance observer assumes that the controlled plant is of the minimum phase and then designs it using the inverse model of the controlled plant. In this study, the proposed disturbance rejection control was designed using parameters obtained through iterative tuning and the model of the plant or the sensitivity function was not required. The entire design process was data driven.

**Remark 5:** To avoid amplifying the measurement noise, filter \( Q \) was designed as a low-pass filter to filter out high-frequency noise. The cutoff frequency of the filter should include the frequency range of the disturbance as much as possible. Therefore, the cutoff frequency of \( Q \) should be selected according to the performance and stability requirements. In general, a high cut-off frequency provides a better anti-disturbance performance but increases noise sensitivity, thereby reducing the system stability. The frequency characteristics of the signal at different stages of motion differ; therefore, the cut-off frequency of filter \( Q \) was designed adaptively via time-frequency analysis. The details are provided in Section IV.

**IV. EXPERIMENT**

The proposed algorithm was assessed using the experimental setup described in Section IV-A. In Section IV-B, the convergence and performance of IV-based IFFT are evaluated. In Section IV-C, the proposed algorithm implemented on the experimental setup is described, and the performance of the algorithm based on reference trajectory changes is validated to prove its effectiveness.

**A. EXPERIMENTAL SETUP**

The experimental linear motor system depicted in Fig.2 was employed to validate the theoretical results and the algorithm proposed herein. The servo control system comprises an upper computer, a motion control card, a motor driver,
and a linear motor NEWPORT IDL225. The position of the linear motor was measured using a Heidenhain linear incremental encoder with a resolution of 20 \( \mu \text{m} \); subsequently, it was interpolated with 2000 intervals to obtain a resolution of 10nm. The motor was driven by a digital servo drive. The proposed algorithm was realized using C language on a PowerPMAC controller. The sampling period of the control system was \( T_s = 0.0001 \text{s} \). The frequency response of plant \( P \) is depicted in Fig.3. The closed-loop servo system was excited using two types of fifth-order point-to-point reference trajectories \( r_1 \) and \( r_2 \), as depicted in Fig.4. The number of setpoints for both in the reference was \( N = 6450 \). The feedback controller used in the following experiments was a linear PD controller formulated as 

\[
C_{fb} = K_p + K_d \frac{1 - e^{-T_s}}{T_s},
\]

where \( K_p = 0.3 \) and \( K_d = 0.0005 \). Based on reference trajectory \( r_1 \), \( r_2 \) and feedback controller \( C_{fb} \), experiments were performed to evaluate the performances of the algorithms.

**B. RESULTS OF THE PARAMETRIZATION CONTROLLER**

In the linear motor, the force ripple is a highly undesirable problem that can seriously deteriorate the servo performance. To clarify the nature of the force ripple better, the force applied by the feedback controller was measured while the motor was moving at a constant velocity. As the acceleration is approximately equal to zero and the friction force is assumed to be constant, any chattering in the applied force compensates for the force ripples [30]. By doing several experiments in the negative and positive directions, the offset owing to the constant friction can be removed, and an estimated force ripple can be obtained, as shown in Fig.5.

The frequency response function of the considered plant is depicted in Fig.3, which shows the rigid-body dynamics at less than 100Hz. The time-frequency analysis of the first trial racking error is shown in Fig.6. As shown, the error characteristics during the acceleration and deceleration phases differed from those during the uniform velocity phase. The dominant frequency during the acceleration and deceler-
M. Cao et al.: Combined Feedforward Control and Disturbance Rejection Control Design for a Wafer Stage

FIGURE 7. Iterative result of $\theta_1$ and $\theta_2$ with reference trajectory $r_1$.

ation phases was within the range of 500Hz. Therefore, the cut-off frequencies of the Q-filter during the acceleration and deceleration phases and the uniform velocity phase were set as 460 and 90Hz, respectively. The second-order low-pass filters are expressed as

$$Q_1 = \frac{0.01726 + 0.03452q^{-1} + 0.01726q^{-2}}{1 - 1.595q^{-1} + 0.6645q^{-2}}$$

$$Q_2 = \frac{0.000768 + 0.001537q^{-1} + 0.000768q^{-2}}{1 - 1.92q^{-1} + 0.9231q^{-2}}$$

where $Q_1$ and $Q_2$ were used for the acceleration and deceleration phases and the uniform velocity phase, respectively.

The basis function $\Psi_{ff}$ was set as $\Psi_{ff} = [\psi_1, \psi_2]$ with

$$\psi_1 = \left(\frac{1 - q^{-1}}{T_s}\right)^2 = \frac{1 - 2q^{-1} + q^{-2}}{T_s^2},$$

$$\psi_2 = \left(\frac{1 - q^{-1}}{T_s}\right),$$

and the corresponding parameter vector as $\theta = [\theta_1, \theta_2]^T$.

Parameterized $C_{ff}(\theta) = \Psi_{ff} \theta$ comprises acceleration feedforward and velocity feedforward, in which $\psi_1 \theta_1$ contributes to the acceleration feedforward compensating for the rigid-body dynamics, whereas $\psi_2 \theta_2$ contributes to the velocity feedforward compensating for viscous friction. Considering the trade-off between the convergence speed and the stability of the algorithm, learning gain $\rho$ was set to 0.8. Estimation results of $\theta_1$ and $\theta_2$ at each iteration are depicted in Fig.7. As shown, their values changed slightly after the 2nd iteration and satisfied the convergence criterion; this implies that the algorithm converged in two steps, thereby confirming that the operating time of the proposed algorithm was acceptable.

C. PERFORMANCE ASSESSMENT OF PROPOSED ALGORITHM

The proposed algorithm, which combined feedforward control and disturbance rejection control, was validated on a wafer stage, as shown in Fig.2.

To verify the effectiveness of the feedforward controller in the proposed IFFT-DRC method, 10 iterations of the experiment were performed. The tracking error at the 10th iteration in comparison with that at the 1st iteration is shown in Fig.8. As shown in Fig.8, the tracking error during the acceleration and deceleration phases reduced significantly. This indicates that the compensation of the feedforward controller effectively eliminated the trajectory-induced error.

To verify the high control precision and strong anti-disturbance capability of the proposed method, comparison experiments with the existing PD-type ILC and IFFT methods were implemented. The updating law of the PD-type ILC is expressed as $u_{k+1} = u_k + L_p e_k + L_d \dot{e}_k$, with $L_p = 0.1$, and $L_d = 0.00005$. The trajectory $r_1$ in Fig.4 was used as the same reference trajectory for the comparison experiment, followed by 10 iterations for ILC and IFFT, separately. The tracking errors and control input signals of the three methods are shown in Fig.9 and Fig.10, respectively. In addition, the mean and root mean square (RMS) error of the three methods...
are shown in Table 1. The following were observed based on Fig. 9 and Table 1:

1. ILC and IFFT eliminated almost all the trajectory-induced errors. However, a heavy fluctuation was observed because these two methods could not eliminate the disturbance-induced error.

2. The control precision of the proposed method was much better than those of the existing ILC and IFFT. The proposed method can eliminate both the trajectory- and disturbance-induced errors.

To verify the high extrapolation capability of the proposed method for tasks, comparison experiments with different reference trajectories were performed. Feedforward control signals and feedforward parameters obtained with \( r_1 \) in the previous comparison experiment were applied to \( r_2 \). Figs. 11 and 12 show the tracking errors and control input signals, respectively, of the three methods under reference trajectory \( r_2 \). The following were observed based on Fig. 11 and Table 1:

1. The performance of the PD-type ILC was highly dependent on the specified reference trajectory. The tracking precision deteriorated severely when the optimal feedforward signal learned by \( r_1 \) was applied to \( r_2 \).

2. The performance of the IFFT and the proposed IFFT-DRC method did not deteriorate when the reference changed, as shown in Fig. 11. Meanwhile, the control precision of the proposed method was much better than that of the IFFT. This confirms that the proposed IFFT-DRC method has a high extrapolation ability for different reference trajectories.

3. Using IFFT to calculate \( \theta^* \) such that the data-based approximation of the system inverse can compensate for the reference-induced error and using \( C_{ff} \) to construct a disturbance observer to compensate for the disturbance-induced error, the algorithm becomes data-driven without requiring a system model.

V. CONCLUSION
Herein, a new algorithm that combined IV-based iterative parameter tuning with disturbance rejection control was proposed to increase the extrapolation ability and stability of varying references tasks. The algorithm is data-driven and does not require a priori knowledge of the system; meanwhile, system inverse is estimated, and disturbance is compensated for with iterative parameter tuning in an iterative process. The experimental results validated the effectiveness of the proposed algorithm. Furthermore, the latter was compared with pre-existing approaches. It was demonstrated that the proposed algorithm achieved (i) high tracking precision, (ii) strong anti-disturbance capability, and (iii) high extrapolation ability in industrial variable trajectory tracking tasks.
REFERENCES

[1] S. Arimoto, S. Kawamura, and F. Miyazaki, “Bettering operation of robots by learning,” J. Robot. Syst., vol. 1, no. 2, pp. 123–140, 1984.

[2] B. T. Fine, S. Mishra, and M. Tomizuka, “Model inverse based iterative learning control using finite impulse response approximations,” in Proc. Amer. Control Conf., 2009, pp. 931–936.

[3] H.-F. Chen and H.-T. Fang, “Output tracking for nonlinear stochastic systems by iterative learning control,” IEEE Trans. Autom. Control, vol. 49, no. 4, pp. 583–588, Apr. 2004.

[4] S. S. Saab, “A discrete-time stochastic learning control algorithm,” IEEE Trans. Autom. Control, vol. 46, no. 6, pp. 876–887, Jun. 2001.

[5] R. Zhang, Z. Hou, R. Chi, and H. Ji, “Adaptive iterative learning control for nonlinearly parameterised systems with unknown time-varying delays and input saturations,” Int. J. Control, vol. 88, no. 6, pp. 1133–1141, Jun. 2015.

[6] Y.-C. Wang, C.-J. Chien, and C.-C. Teng, “An adaptive PID-type iterative learning controller for unknown nonlinear systems with initial state errors,” in Proc. Eur. Control Conf., Sep. 2003, pp. 607–612.

[7] D. Meng and K. L. Moore, “Robust iterative learning control for non-repetitive uncertain systems,” IEEE Trans. Autom. Control, vol. 62, no. 2, pp. 907–913, Feb. 2017.

[8] M.-T. Lin, C.-L. Yen, M.-S. Tsai, and H.-T. Yau, “Application of robust adaptive learning algorithm in motion control system,” Mechatronics, vol. 22, no. 5, pp. 530–540, Aug. 2013.

[9] F. Bouakrif and M. Zasadzinski, “Trajectory tracking control for perturbed robot manipulators using iterative learning method,” Int. J. Adv. Manuf. Technol., vol. 87, nos. 5–8, pp. 2013–2022, Nov. 2016.

[10] S. Mishra and M. Tomizuka, “Segmented iterative learning control for precision positioning of wafer stages,” in Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics, Sep. 2007, pp. 1–6.

[11] M. Li, Y. Zha, K. Yang, and C. Hu, “A data-driven variable-gain control strategy for an ultra-precision wafer stage with accelerated iterative parameter tuning,” IEEE Trans. Ind. Informat., vol. 11, no. 5, pp. 1179–1189, Aug. 2015.

[12] J. A. Butterworth, L. Y. Pao, and D. Y. Abramovich, “Analysis and comparison of three discrete-time feedforward model-inverse control techniques for nonminimum-phase systems,” Mechatronics, vol. 22, no. 5, pp. 577–587, Aug. 2012.

[13] T. Yang, N. Sun, H. Chen, and Y. Fang, “Observer-based nonlinear control for tower cranes suffering from uncertain friction and actuator constraints with experimental verification,” IEEE Trans. Ind. Electron., early access, May 12, 2020, doi: 10.1109/TIE.2020.2992972.

[14] Y. Wu, N. Sun, H. Chen, and Y. Fang, “Adaptive output feedback control for 5-DOF varying-cable-length tower cranes with cargo mass estimation,” IEEE Trans. Ind. Informat., early access, Jun. 30, 2020, doi: 10.1109/TII.2020.3006179.

[15] N. Sun, Y. Fu, T. Yang, J. Zhang, Y. Fang, and X. Xin, “Nonlinear motion control of complicated dual rotary crane systems without velocity feedback: Design, analysis, and hardware experiments,” IEEE Trans. Autom. Sci. Eng., vol. 17, no. 2, pp. 1017–1029, Apr. 2020.

[16] K.-S. Kim and Q. Zou, “A modeling-free inversion-based iterative feedforward control for precision output tracking of linear time-invariant systems,” IEEE/ASME Trans. Mechatronics, vol. 18, no. 6, pp. 1767–1777, Dec. 2013.

[17] J. van de Wijdeven and O. H. Bosgra, “Using basis functions in iterative learning control: Analysis and design theory,” Int. J. Control, vol. 83, no. 4, pp. 661–675, Apr. 2010.

[18] M. Li, Y. Zhu, K. Yang, L. Yang, C. Hu, and H. Mu, “Convergence rate oriented iterative feedback tuning with application to an ultraprecision wafer stage,” IEEE Trans. Ind. Electron., vol. 66, no. 3, pp. 1993–2003, Mar. 2019.

[19] P. C. Young, “Refined instrumental variable estimation: Maximum likelihood optimization of a unified Box–Jenkins model,” Automatica, vol. 52, pp. 35–46, Feb. 2015.

[20] F. Boeren, D. Bruijnien, N. van Dijk, and T. Oomen, “Joint input shaping and feedforward for point-to-point motion: Automated tuning for an industrial nanopositioning system,” Mechatronics, vol. 24, no. 6, pp. 572–581, Sep. 2014.

[21] F. Boeren, T. Oomen, and M. Steinbuch, “Iterative motion feedforward tuning: A data-driven approach based on instrumental variable identification,” Control Eng. Pract., vol. 37, pp. 11–19, Apr. 2015.

[22] F. Boeren, D. Bruijnien, and T. Oomen, “Enhancing feedforward controller tuning via instrumental variables: With application to nanopositioning,” Int. J. Control, vol. 90, no. 4, pp. 746–764, Apr. 2017.

[23] M. Gilson, H. Garnier, P. C. Young, D. H. Van, and P. M. J. Van den Hof, “Optimal instrumental variable method for closed-loop identification,” IET Control Theory Appl., vol. 5, no. 10, pp. 1147–1154, 2011.

[24] F. Song, Y. Liu, J.-X. Xu, X. Yang, and Q. Zhu, “Data-driven iterative feedforward tuning for a wafer stage: A high-order approach based on instrumental variables,” IEEE Trans. Ind. Electron., vol. 66, no. 4, pp. 3106–3116, Apr. 2019.

[25] L. Blanken, F. Boeren, D. Bruijnien, and T. Oomen, “Batch-to-batch rational feedforward control: From iterative learning to identification approaches, with application to a wafer stage,” IEEE/ASME Trans. Mechatronics, vol. 22, no. 2, pp. 826–837, Apr. 2017.

[26] J. Bolder, J. van Zundert, S. Koekoekbakker, and T. Oomen, “Enhancing flatbed printer accuracy and throughput: Optimal rational feedforward controller tuning via iterative learning control,” IEEE Trans. Ind. Electron., vol. 64, no. 5, pp. 4207–4216, May 2017.

[27] W. Chen, J. Yang, L. Guo, and S. Li, “Disturbance-observer-based control and related methods—An overview,” IEEE Trans. Ind. Electron., vol. 63, no. 2, pp. 1083–1095, Feb. 2016.

[28] J. Han, “From PID to active disturbance rejection control,” IEEE Trans. Ind. Electron., vol. 56, no. 3, pp. 900–906, Mar. 2009.

[29] A. H. M. Sayem, Z. Cao, and Z. Man, “Model-free ESO-based repetitive control for rejecting periodic and aperiodic disturbances,” IEEE Trans. Ind. Electron., vol. 64, no. 4, pp. 3433–3441, Apr. 2017.

[30] M. Butcher and A. Karimi, “Linear parameter-varying iterative learning control with application to a linear motor system,” IEEE/ASME Trans. Mechatronics, vol. 15, no. 3, pp. 412–420, Jun. 2010.