

## On-line instrumental variable-based feedforward tuning for non-resetting motion tasks

Mooren, Noud; Witvoet, Gert; Oomen, Tom

**DOI**

[10.1002/rnc.6925](https://doi.org/10.1002/rnc.6925)

**Publication date**

2023

**Document Version**

Final published version

**Published in**

International Journal of Robust and Nonlinear Control

**Citation (APA)**

Mooren, N., Witvoet, G., & Oomen, T. (2023). On-line instrumental variable-based feedforward tuning for non-resetting motion tasks. *International Journal of Robust and Nonlinear Control*, 33(18), 11000-11018. <https://doi.org/10.1002/rnc.6925>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

## RESEARCH ARTICLE

WILEY

# On-line instrumental variable-based feedforward tuning for non-resetting motion tasks

Noud Mooren<sup>1</sup>  | Gert Witvoet<sup>1,2</sup> | Tom Oomen<sup>1,3</sup>

<sup>1</sup>Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands

<sup>2</sup>Optomechatronics Department, TNO, Delft, The Netherlands

<sup>3</sup>Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands

## Correspondence

Noud Mooren, Department of Mechanical Engineering, Eindhoven University of Technology, Postbus 513, 5600 Eindhoven, The Netherlands.

Email: [n.f.m.mooren@tue.nl](mailto:n.f.m.mooren@tue.nl)

## Funding information

European Union H2020, Grant/Award Number: 637095; ECSEL-2016-1, Grant/Award Numbers: 737453, 101007311

## Summary

Data-driven feedforward control for tracking of varying and non-resetting point-to-point references requires continuous updating of feedforward parameters instead of task-by-task updating. The aim of this paper is to develop an adaptive feedforward controller for non-resetting point-to-point motion tasks by a data-driven feedforward controller. An approximate optimal instrumental variable (IV) estimator with real-time bootstrapping is employed in a closed-loop setting to update the feedforward parameters. A case study on a wafer-stage and experimental validation on a benchmark motion system show the performance benefit.

## KEYWORDS

adaptive feedforward, instrumental variables, parameter estimation, reference tracking

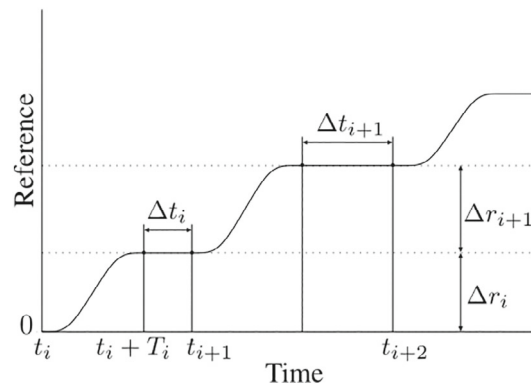
## 1 | INTRODUCTION

Data-driven feedforward control can yield tremendous performance improvement for high-precision motion systems that require highly flexible and non-resetting motion tasks as in Figure 1. Consider for example the wafer-stage for semiconductor manufacturing in Figure 2, which has a short-stroke stage (blue) for accurate and fast positioning of the semiconductor components in front of a gripper, and a long-stroke stage (yellow) to enable a large range. Sequentially picking up semiconductor components from the wafer constitutes such a varying and non-resetting point-to-point reference in Figure 1, where  $\Delta r_i$  varies depending on the relative position between the current and the next semiconductor component, and the dwell time  $\Delta t_i$  depends on other modules in the system to be ready for the next task, see, for example, References 1–3. Moreover, the system dynamics are position-dependent due to the large range. Hence, a data-driven feedforward approach is required that can cope with continuous motion tasks with a varying step size and task length, that is, the system does not resets to the same initial condition at the start of a next task which is required for batch-wise feedforward approaches.<sup>4</sup> As a result, batch-wise learning of feedforward parameters is not possible and does not exploit data during the current task.

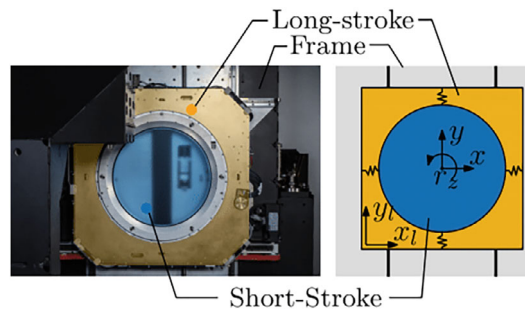
Manual tuning of feedforward parameters and model-based feedforward can compensate for non-resetting and varying references, that is, with an inverse model of the system any reference can be completely compensated. However, the performance is directly related to the inverse-model quality, and manual tuning of feedforward parameters,<sup>5</sup> for example, mass, snap and jerk parameters, or non-linear phenomena such as hysteresis,<sup>6</sup> is often time-consuming and need to be

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial-NoDerivs](https://creativecommons.org/licenses/by-nc-nd/4.0/) License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

© 2023 The Authors. *International Journal of Robust and Nonlinear Control* published by John Wiley & Sons Ltd.



**FIGURE 1** Example of a typical point-to-point non-resetting reference with varying step size  $\Delta r_i \in \mathbb{R}$  and dwell time  $\Delta t_i \in \mathbb{N}$ .



**FIGURE 2** Motivating application: A long-stroke short-stroke wafer-stage for semiconductor manufacturing that performs non-resetting and varying point-to-point motion tasks.

performed on an inefficient machine-specific basis. Moreover, in the case of position-dependent behavior, a different feedforward tuning is required over the entire range. In view of these challenges, an automatic data-driven tuning approach is preferred to learn feedforward controllers.

Batch-wise learning approaches, such as iterative learning control (ILC), learn from previous tasks to improve tracking performance for systems with resetting references. Iterative learning control (ILC) learning a feedforward signal that exactly compensates the same reference, even in the case of a non-perfect model.<sup>4,7,8</sup> If varying and resetting tasks appear, then ILC can be combined with basis function and learn the corresponding parameters.<sup>9,10</sup> However, as ILC requires the same initial condition at the start of each task  $r_i$  this batch-wise approach is not applicable<sup>11</sup> if the reference is non-resetting as in Figure 1.

On-line learning of feedforward parameters enables performance improvements within a task without imposing restrictions on the resetting behavior of the reference. In Reference 12 recursive least squares (RLS) is successfully used for on-line estimation of the acceleration feedforward parameter on a wafer-stage resulting in an immediate performance improvement. However, in Reference 13 it is shown that this approach inherently suffers from a closed-loop estimation problem resulting in biased estimates resulting in performance degradation. For accurate reference tracking, it is essential that unbiased parameter estimates in an on-line setting.

Instrumental variable (IV) estimators can obtain unbiased estimates in a closed-loop setting by appropriate design of so-called instrumental variables, see, for example, References 14–20. In Reference 21 an IV estimator is applied for optimal updating of feedforward parameters. This result shows that for rational systems with resetting point-to-point motion tasks, a combination of an adaptive input-shaper and feedforward-controller facilitates a convex optimization problem with an analytic solution. However, IV estimation is performed in a batch-wise setting it is not applicable to non-resetting varying references.

Although recent progress has been made for on-line feedforward controller tuning for non-resetting tasks, an approach that yields an accurate estimate, that is, unbiased with small variance, in a closed-loop setting is not yet available. The main contribution of this paper is an optimal IV-based adaptive feedforward controller for non-resetting point-to-point

references with an on-line bootstrapping approach,<sup>14,22,23</sup> yielding a convex optimization problem for efficient on-line implementation. The following contributions are identified.

- C1 IV estimation of feedforward parameters for arbitrary non-resetting point-to-point references,
- C2 recursive IV estimation through on-line bootstrapping,
- C3 a case study on the wafer-stage in Figure 2 with non-resetting varying point-to-point references, and
- C4 experimental validation on benchmark motion system.

The outline of this paper is as follows. In Section 2, the control problem is formulated as an estimation problem that is formally presented. In Section 3, the recursive IV-based estimator for on-line feedforward parameter estimation is presented with an on-line bootstrapping approach. In Section 4, additional design aspects are outlined. In Section 5, a simulation case-study is performed with a wafer-stage setup, and in Section 6 the approach is validated on a benchmark motion system. Finally, in Section 7, the conclusions are presented.

## 2 | PROBLEM DEFINITION

### 2.1 | Control setting and problem definition

The control setting is depicted in Figure 3, where  $P$  is a linear time-invariant (LTI) single-input single-output (SISO) system that represent the dynamics of the short-stroke wafer-stage in the  $x$ -direction, and  $C_{fb}$  is a stabilizing LTI feedback controller. The system  $P$  is rational given by

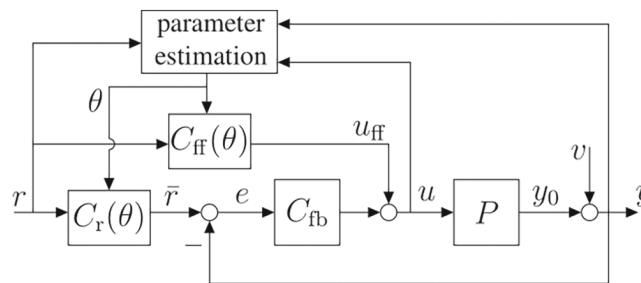
$$P = \frac{B_0(q^{-1}, \theta_0)}{A_0(q^{-1}, \theta_0)}, \quad (1)$$

where  $A_0$  and  $B_0$  are polynomials in the backwards time-shift operator  $q^{-1}$ , that is,  $u(k-1) = q^{-1}u(k)$  with  $k \in \mathbb{Z}$  discrete time, and  $\theta_0 \in \mathbb{R}^{n_\theta}$  are the true system parameters. The aim is to track a non-resetting point-to-point reference  $r$  depicted in Figure 1, by designing the adaptive feedforward controller  $C_{ff}(\theta)$  and input shaper  $C_r(\theta)$  that depend on  $\theta(k)$ . The signal  $\theta(k) \in \mathbb{R}^{n_\theta}$  contains  $n_\theta \in \mathbb{N}$  parameters that are updated in an on-line fashion by a parameter estimator on the basis of  $r(k)$ , the system input  $u(k)$  and the noisy output  $y(k) = y_0(k) + v(k)$ , where  $v(k) \sim \mathcal{N}(0, \sigma_v^2)$  is zero-mean normally distributed with variance  $\sigma_v^2$ . By updating the feedforward parameters adaptively, essentially an additional feedback loop is closed around the feedforward controller. In the remainder, the focus is on improving performance in the dwell time phase  $\Delta t_i \in \mathbb{N}$ , for example, where the semi-conductor component in the earlier example is picked-up from the wafer.

**Definition 1.** The control goal is to minimize the reference tracking error  $\|e_r\|^2$  with  $e_r = r - y$ , during the dwell time  $\Delta t_i$  by designing the adaptive feedforward controller  $C_{ff}(\theta)$  and input shaper  $C_r(\theta)$ .

The tracking error  $e_r$  to be minimized is given by

$$e_r = (1 - SP(C_{ff}(\theta_k) + C_{fb}C_r(\theta_k)))r, \quad (2)$$



**FIGURE 3** Control problem with adaptive input shaper  $C_r$  and feedforward controller  $C_{ff}$  with parameter  $\theta(k)$  this is estimated in real-time.

where  $S = (1 + PC_{fb})^{-1}$  is the sensitivity function and  $\theta_k = \theta(k)$ . It is important to note that the error (2) is not necessarily equal to the feedback error  $e = \bar{r} - y$ , with  $\bar{r} = C_r(\theta)r$ , which is minimized by  $C_{fb}$ .

**Remark 1.** The feedback controller  $C_{fb}$  is assumed to be LTI and stabilizing, furthermore, the design of  $C_{fb}$  does not influence the design of the algorithm provided in this paper. Therefore, the design of  $C_{fb}$  is considered out-of-the-scope in the remainder of this article.

## 2.2 | Feedforward controller parameterization

The input-shaper  $C_r$  and feedforward controller  $C_{ff}$  are polynomials and linear in the parameters  $\theta$ , which facilitate a convex feedforward parameter estimation problem that is beneficial for the on-line implementation. Consider the following parameterization, see also Reference 3.

**Definition 2.** The input shaper  $C_r(\theta)$  and feedforward controller  $C_{ff}(\theta)$  are of the form

$$C = \left\{ (C_r, C_{ff}) \left| \begin{array}{l} C_{ff} = A(q^{-1}, \theta_k) \\ C_r = B(q^{-1}, \theta_k) \end{array} \right. \right\}, \quad (3)$$

with  $A(q^{-1}, \theta_k)$  and  $B(q^{-1}, \theta_k)$  polynomials in  $q^{-1}$  given by,

$$A(q^{-1}, \theta_k) = \sum_{i=1}^{n_a} \psi^i(q^{-1})^i \theta_k^i = \Psi_A^T \theta_A, \quad \text{and}, \quad (4)$$

$$B(q^{-1}, \theta_k) = 1 + \sum_{i=n_a+1}^{n_a+n_b} \psi^i(q^{-1})^i \theta_k^i = 1 + \Psi_B^T \theta_B, \quad (5)$$

which are linear in  $\theta_k$ . Where  $\psi(q^{-1}) \in \mathbb{R}[q^{-1}]$  are the basis functions where  $n_b, n_a \in \mathbb{N}$ , and with the following notation

$$\Psi_A = \begin{bmatrix} \psi^1 & \dots & \psi^{n_a} \end{bmatrix}^T \in \mathbb{R}^{n_a}[q^{-1}] \quad (6)$$

$$\Psi_B = \begin{bmatrix} \psi^{n_a+1} & \dots & \psi^{n_a+n_b} \end{bmatrix}^T \in \mathbb{R}^{n_b}[q^{-1}] \quad (7)$$

these are basis functions of  $C_{ff}$  and  $C_r$  respectively with corresponding parameters  $\theta_A = [\theta^1 \dots \theta^{n_a}]^T \in \mathbb{R}^{n_a}$ ,  $\theta_B = [\theta^{n_b} \dots \theta^{n_a+n_b}]^T \in \mathbb{R}^{n_b}$ , and  $\theta(k) = [\theta_A(k) \quad \theta_B(k)]^T \in \mathbb{R}^{n_\theta}$  with  $n_\theta = n_a + n_b$ .

The following assumption is imposed on the input shaper.

**Assumption 1.** The input shaper  $C_r$  satisfies that

$$C_r(\theta, q^{-1})|_{q^{-1}=1} = 1, \quad (8)$$

such that it has unit steady-state gain.

Assumption 1 avoids scaling of the reference. Note that the reference  $\bar{r}(k) = C_r r(k)$  reaches steady state  $n_k$  samples after  $r(k)$  is constant which is the case in the dwell time  $\Delta t$ , with  $n_k$  the order of  $C_r$ . Consequently,  $\bar{r}(k) = r(k)$  for  $k \in \Delta T$  with  $\Delta T = [t_i + T_i + n_k, t_{i+1}] \forall i \in \mathbb{N}$ , see, for example, Reference 3. As a result, the tracking error (2) is identical to the feedback error

$$e = S(C_r(\theta) - PC_{ff}(\theta))r, \quad (9)$$

for all  $k \in \Delta T$ , minimization of (9) is obtained if, ideally,

$$C_r(\theta) = PC_{ff}(\theta) \quad (10)$$

for some  $\theta$  such that  $C_{ff}C_r^{-1} = P^{-1}$  contains a model of the inverse system.

### 2.3 | Estimation for feedforward control

In this section, the control goal in Definition 1 is reformulated as an equivalent estimation problem under Assumption 1 and the feedforward controller parameterization  $C$ . The ideal setting in (10) implies that  $e = 0$  if  $C_r(\theta) = A_0$  and  $C_{ff}(\theta) = B_0$  for arbitrary references. Hence, it is necessary that the true system  $P$  must be in the model set that is defined by the basis functions  $\Psi_A$  and  $\Psi_B$ .

**Assumption 2.** The true system is in the model set, that is, the basis functions  $\Psi_A$  and  $\Psi_B$  satisfy that  $\Psi_A\theta_A = A_0$ , and  $\Psi_B\theta_B = B_0$  for some  $\theta_A$  and  $\theta_B$ .

The following definition outlines properties of the parameter estimator such that the tracking error is minimized during the dwell time  $\Delta t$ .

**Definition 3.** In the setting in Figure 3 under Assumptions 1 and 2, and with the parameterization  $C$ , the tracking error  $e_r(k)$  for  $k \in \Delta T$  is minimized by obtaining an accurate estimate  $\theta(k)$ , i.e., such that the estimation error  $\theta(k) - \theta_0$  asymptotically has the normal distribution

$$\sqrt{k}(\theta - \theta_0) \rightarrow \mathcal{N}(0, P_{\text{cov}}) \quad \text{for } k \rightarrow \infty, \quad (11)$$

with the following properties:

- R1  $(\theta(k) - \theta_0) \rightarrow 0$  as  $k \rightarrow \infty$ ; and
- R2  $P_{\text{cov}}$  is minimal.

Definition 3 yields that by design of an appropriate estimator, that is, that recovers the true plant parameters in the presence of noise  $v$  and in a closed-loop setting, the tracking error is minimized during the dwell time  $\Delta t_i$ .

### 2.4 | Control problem

Estimation of the feedforward controller parameters is in practice often done using a least-squares estimator, see, for example, References 12, 24. However, it is shown that biased estimates are often obtained if the estimation is performed in a closed-loop setting with measurement noise, see, for example, Reference 13. The aim of this paper is to use a suitable estimator, in view of Definition 3, to estimate the feedforward parameters in the closed-loop setting in Figure 3 with  $v \neq 0$ .

## 3 | ON-LINE IV-BASED FEEDFORWARD CONTROLLER TUNING

In this section, an approximate optimal IV estimator with on-line bootstrapping is presented that yields a recursive update law for the feedforward parameters  $\theta(k)$  in the setting in Figure 3, which consequently minimizes the tracking error during the dwell time, see Sections 3.1 and 3.3. This includes conditions on the optimal instrumental variables to obtain the theoretical lower bound, which appears to depend on the true system parameters. Therefore, an on-line bootstrapping procedure is presented that approximates the optimal instrumental variables in practice in Section 3.2. Finally, an implementation procedure is provided in Section 3.4.

### 3.1 | Optimal IV estimation for feedforward

A suitable estimator must be developed for the adaptive feedforward parameters  $\theta(k)$  as shown in Definition 3 to minimize the tracking error. An IV estimator is used that enables recursive tuning of  $\theta(k)$  on the basis  $u(k)$ ,  $y(k)$  and  $r(k)$ . Moreover, it is shown that the presented IV estimation problem yields a recursive update law for the feedforward parameters  $\theta(k)$  that is suitable for on-line implementation.

**Theorem 1.** Consider the setup of Figure 3 with feedforward parameterization  $C$  in (3), and under Assumptions 1 and 2. The feedforward parameters that satisfy R1 and R2, and consequently minimize the  $e_r(k)$  for  $k \in \Delta T$ , are obtained by

$$\theta(k) = \arg \min_{\theta} V_{IV}(k, \theta), \quad (12)$$

with

$$V_{IV}(k, \theta) = \left\| \left[ \frac{1}{k} \sum_{i=1}^k z(i) (F(q^{-1})\epsilon(i, \theta)) \right] \right\|^2, \quad (13)$$

$$\epsilon(k, \theta) = u(k) - \phi^T(k)\theta(k), \quad (14)$$

where  $\epsilon(k, \theta) \in \mathbb{R}$  is linear in  $\theta$ ,  $z(k) \in \mathbb{R}^{n_z}$ ,  $F(q^{-1})$  is a stable filter, and

$$\phi(k) = \begin{bmatrix} \Psi_A(q^{-1})y(k) \\ -\Psi_B(q^{-1})u(k) \end{bmatrix} \in \mathbb{R}^{n_{\theta}}. \quad (15)$$

The estimate (12) is consistent, that is,  $(\theta_0 - \theta(k)) \rightarrow 0$  for  $k \rightarrow \infty$ , if

C1  $\mathbb{E}[z(k)F(q^{-1})A_0(\theta_0)v(k)] = 0$ , that is, such that the noise  $v$  is uncorrelated with the instrumental variables  $z$ ,

C2  $\mathbb{E}[z(k)F(q^{-1})\phi(k)]$  is non-singular, that is, the regressor  $\phi$  and the instruments  $z$  must be correlated, and

the estimator error has optimal asymptotic distribution  $(\theta(k) - \theta_0) \rightarrow \mathcal{N}(0, P_{IV})$  with minimal variance if

$$z(k) = A_0^{-1}(q^{-1}, \theta_0)\phi_r(k) := z^{\text{opt}}(k) \quad (16)$$

$$F(q^{-1}) = A_0^{-1}(q^{-1}, \theta_0) := F^{\text{opt}}(q^{-1}), \quad (17)$$

where

$$\phi_r(k) = \begin{bmatrix} \Psi_A(q^{-1})y_r(k) \\ -\Psi_B(q^{-1})u_r(k) \end{bmatrix}, \quad (18)$$

is the noise-free part of  $\phi(k)$ , that is, with  $v = 0$ , and  $u_r(k)$  and  $y_r(k)$  are the noise-free input and output respectively.

The proof of Theorem 1 is included in Appendix.

In this paper, a recursive IV-based procedure is presented that exploits on-line bootstrapping to closely approximate optimal IV. Under optimality conditions, the result in Theorem 1 yields that the plant parameters are consistently recovered with minimal variance. Moreover, in the non-optimal case the parameter estimates remain unbiased but the variance can be larger than the theoretical lower bound. As a result, the reference-induced tracking error (9) is minimized. The presented approach updates each sample in contrast to the existing approaches, for example, Reference 25 which exploits resetting reference tasks. In the case of repeating reference tasks the batch-wise approach in Reference 25 can be used,



where bootstrapping is avoided as shown later. Alternatively, one can use a regularized LS estimation to minimize variance at the cost of introducing a bias.

Next, consider the following persistence of excitation condition.

**Definition 4.** The input  $u$  is persistently exciting of order  $n$  if for all  $k$  there exists an integer  $m$  such that

$$\gamma_1 I > \sum_{i=k}^{k+m} \xi(i) \xi(i)^\top > \gamma_2 I, \quad (19)$$

where  $\gamma_1, \gamma_2 > 0$ , and  $\xi(k) = [u(k-1) \ \dots \ u(k-n)]$  see [chap. 2.4; References 26, 27].

**Assumption 3.** The reference is persistently exciting according to Definition 4.

*Remark 2.* In practice, Assumption 3 can be satisfied by ensuring that the order of the reference or the feedback controller  $C_{fb}$ , which influence the input  $u$ , are sufficiently high.

Theorem 1 reveals that with the IV-estimator (12) the estimate  $\theta(k) \rightarrow \theta_0$  for  $k \rightarrow \infty$  if the instrumental variables  $z(t)$  are uncorrelated with the noise  $v(t)$  (C1) and correlated with the regressor (C2), satisfying R1. Moreover, with the optimal instrumental variables  $z^{\text{opt}}$  in (16) and the optimal pre-filter  $F^{\text{opt}}$  in (17), the theoretical lower-bound of the covariance matrix  $P_{IV}$  is obtained, satisfying R2. However, since both  $z^{\text{opt}}$  and  $F^{\text{opt}}$  depend on the unknown system parameters  $\theta_0$ , approximations of the optimal instrumental variables and pre-filter are provided next with on-line bootstrapping. Finally, under Assumptions 2 and 3 and with the estimator in Theorem 1 the estimate is said to be identifiable.<sup>14</sup>

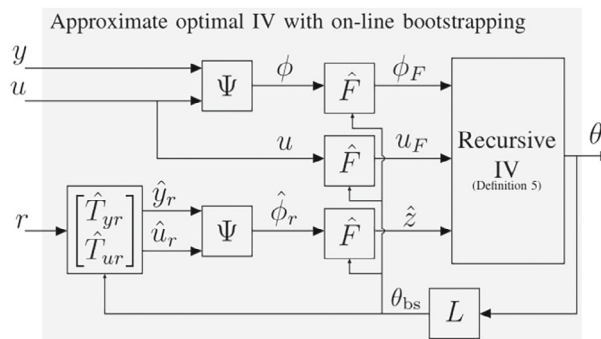
Note that C1 directly imposes the limitation that  $z(t)$  cannot be constructed from measured signals that contain noise. Therefore, an external signal is required; in this work the reference is used to construct the instrumental variables, that is,  $z(t) = f(r(t), \dot{r}(t), \ddot{r}(t), \dots)$ , as in, for example, Reference 21.

### 3.2 | Towards optimal feedforward using on-line bootstrapping

In this section, an approximation of the optimal instruments  $z^{\text{opt}}$  and pre-filter  $F^{\text{opt}}$  is provided using on-line bootstrapping to approximate the optimal variance, see, for example, References 14, 22, 23 for more details on bootstrapping. This is done by replacing  $\theta_0$  by an estimate denoted  $\theta_{bs}(k)$  at each sample. The aim is to approximate the theoretical lower-bound of the estimator variance. The approach is presented in Figure 4, where  $L$  is an optional pre-filter, and  $\hat{z}$  and  $\hat{F}$  are approximation of their optimal versions. In this section, set  $L(q^{-1}) = q^{-1}$ , such that  $\theta_{bs}(k) = \theta(k-1)$  is the previous estimate.

First, the approximation of the optimal pre-filter in (17) is straight forward by replacing  $\theta_0$  with  $\theta_{bs}(k)$  yielding

$$\hat{F}(q^{-1}, \theta_{bs}, k) = (\Psi_A(q^{-1}) \theta_{bs}^A(k))^{-1}. \quad (20)$$



**FIGURE 4** Schematic representation for approximate optimal instrumental variable (IV) estimation for adaptive feedforward with on-line bootstrapping, where  $\Psi = [\Psi_A \ -\Psi_B]^\top$ .



Second, the approximation of  $z^{\text{opt}}$  in (16) is more involved, because it requires the noise-free regressor  $\phi_r(k)$  in (18) that requires  $y_r$  and  $u_r$ . To approximate  $\phi_r(k)$ , note that  $u_r$  and output  $y_r$  can be written as function of  $r(k)$ , i.e.,

$$y_r(k) = T_{yr}(\theta_0)r(k) \quad (21)$$

$$u_r(k) = T_{ur}(\theta_0)r(k), \quad (22)$$

with

$$T_{yr}(\theta_0) = S(\theta_0)C(\theta) \quad (23)$$

$$T_{ur}(\theta_0) = P(\theta_0)S(\theta_0)C(\theta) \quad (24)$$

$$C(\theta) = C_{\text{ff}}(\theta) + C_{\text{fb}}C_r(\theta), \quad (25)$$

here,  $T_{ur}(q^{-1}, \theta_0)$  and  $T_{yr}(q^{-1}, \theta_0)$  are the mappings from  $r$  to the noise-free input and output respectively. Since  $T_{ur}$  and  $T_{yr}$  also depend on  $\theta_0$ , a similar bootstrapping approach is utilized, i.e.,

$$\hat{u}_r(k) = \hat{T}_{ur}(\theta_{\text{bs}}(k))r(k) \quad (26)$$

$$\hat{y}_r(k) = \hat{T}_{yr}(\theta_{\text{bs}}(k))r(k) \quad (27)$$

are used as the approximations of the noise-free input and output with

$$\hat{T}_{yr}(q^{-1}, \theta_{\text{bs}}) = S(\theta_{\text{bs}})C(\theta) \quad (28a)$$

$$\hat{T}_{ur}(q^{-1}, \theta_{\text{bs}}) = P(\theta_{\text{bs}})S(\theta_{\text{bs}})C(\theta). \quad (28b)$$

*Remark 3.* The filters (28) can become unstable for some  $\theta_{\text{bs}}$ , in which case a stable approximation is presented in Section 4. Moreover, a practical design choice for the optional filter  $L$  is presented in Section 4 to minimize potential divergence of  $\theta_{\text{bs}}$ .

Consequently, the noise-free regressor is given by

$$\hat{z}(k) = \begin{bmatrix} \Psi_A \hat{T}_{yr}(\theta_{\text{bs}}) & \Psi_B \hat{T}_{ur}(\theta_{\text{bs}}) \end{bmatrix} r(k). \quad (29)$$

In summary, the optimal instruments and the pre-filter are approximated with (29) and (20) respectively, where the parameters  $\theta_{\text{bs}}(k)$  are (filtered versions) of  $\theta(k-1)$  which is referred to as on-line bootstrapping. In the following section, the resulting update law for the feedforward parameters  $\theta$  is outlined.

*Remark 4.* If the reference is resetting, that is, can be split into individual batches, then the parameterized filters (28) can be replaced by a stable approximation of  $C^{-1}$ , see Reference 25. However, that approach is limited to SISO systems which is not the case for the presented approach with bootstrapping.

### 3.3 | Recursive IV for on-line feedforward control

A recursive solution to the parameter estimation problem in Theorem 1 that is suitable for the on-line tuning of  $C_r(\theta)$  and  $C_{\text{ff}}(\theta)$  is presented in the following Definition.

**Definition 5.** The parameters  $\theta(k)$  in  $C_{ff}(\theta)$  and  $C_r(\theta)$  at sample  $k$  are given by the recursive solution to (12) on the basis of  $u(k)$ ,  $\phi(k)$ ,  $z(k)$ , and the (initial) parameter estimate  $\theta(k-1)$ , with the update law

$$\theta(k) = \theta(k-1) + K(k)(v(k) - \Phi^T(k)\theta(k-1)), \quad (30)$$

with

$$\begin{aligned} K(k) &= P(k-1)\Phi(k)[\Lambda(k) + \Phi^T(k)P(k-1)\Phi(k)]^{-1} \\ \Phi(k) &= \begin{bmatrix} R^T(k-1)z(k) & \phi_F(k) \end{bmatrix} \in \mathbb{R}^{n_\theta \times 2} \\ \Lambda(k) &= \begin{bmatrix} -z^T(k)z(k) & 1 \\ 1 & 0 \end{bmatrix} \in \mathbb{R}^{2 \times 2} \\ v(k) &= \begin{bmatrix} z^T(k)\eta(k-1) \\ u_F(k) \end{bmatrix} \in \mathbb{R}^2 \\ R(k) &= R(k-1) + z(k)\phi_F^T(k) \\ \eta(k) &= \eta(k-1) + z(k)u_F(k) \\ P(k) &= P(k-1) - K(k)\Phi^T(k)P(k-1), \end{aligned}$$

where  $K(k) \in \mathbb{R}^{n_\theta \times 2}$ ,  $P(k) \in \mathbb{R}^{n_\theta \times n_\theta}$ ,  $\eta(k) \in \mathbb{R}^{n_z}$ , and  $R(k) \in \mathbb{R}^{n_z \times n_\theta}$  are recursively computed, and  $u_F(k) = F(q^{-1})u(k)$  and  $\phi_F(k) = F(q^{-1})\phi(k)$ , constituting the recursive solution to the IV estimate in Theorem 1.

The update law for the feedforward parameters (30) is used to update the feedforward controllers at each sample. Hence, the control input  $u_{ff}$  and the reference  $\bar{r}$  at sample  $k$  are given by

$$u_{ff}(k) = \Psi_A(q^{-1})\theta_A(k)r(k) \quad (31)$$

$$\bar{r}(k) = \Psi_B(q^{-1})\theta_B(k)r(k), \quad (32)$$

where  $\theta(k)$  is computed using Definition 5.

To compute  $\theta(k)$ , only the previous estimate  $\theta(k-1)$ , the current value of the input  $u(k)$ , the output  $y(k)$  and the instruments  $z(k)$  are required, that is, the recursive algorithm does not require memory of all the past samples to compute the parameter update for the current time step. Moreover, the matrix inversion to compute  $K(k)$  is of size  $2 \times 2$  for all  $k$  which is computationally inexpensive, this allows to implement the IV-based feedforward controller in an on-line setting.

*Remark 5.* If the reference can be separated in individual tasks that start from the same initial conditions, then the batch-wise solution to the identification problem in Theorem 1 is given by

$$\theta_i = (R_N^T R_N)^{-1} R_N^T U_N,$$

for task  $i$  with  $N$  the number of data points and  $R_N = \sum_{i=1}^N z(i)F(q^{-1})\phi^T(i)$  and  $U_N = \sum_{i=1}^N z(i)F(q^{-1})u(i)$ , see, for example, Reference 28.

### 3.4 | Procedure

The following procedure outlines the implementation for approximate optimal IV-based feedforward control.

This procedure completes the basic implementation of on-line bootstrapping IV for adaptive feedforward. In Section 4 additional implementation aspects are provided, for example, how to ensure stability of the filter  $T_{ur}(\theta_{bs})$  and  $T_{yr}(\theta_{bs})$ .

**Procedure 1.** On-line approximate bootstrapping IV**Step 1:** Initialization

- (a) Set  $\theta(0) = \theta_{\text{init}}$ ,  $P(0) = \alpha I_{n_\theta}$  with  $\alpha$  a large number,  $\eta(0) = \underline{0}$  and  $R(0) = \underline{0}$  with  $\underline{0}$  a matrix of appropriate size, and  $k = 1$ . Moreover, define  $\Psi_A(q^{-1})$  and  $\Psi_B(q^{-1})$ .

**Step 2:** On-line IV-based feedforward

- (a) Compute  $\hat{F}^{\text{opt}}(\theta_{\text{bs}})$ ,  $\hat{T}_{\text{yr}}(\theta_{\text{bs}})$  and  $\hat{T}_{\text{ur}}(\theta_{\text{bs}})$  in (28) with  $\theta_{\text{bs}}(k) = L(q^{-1})\theta(k-1)$ .  
 (b) Obtain  $u(k)$ ,  $y(k)$  and  $r(k)$ , compute  $\phi(k)$  in (15), and  $z(k)$ .  
 (c) Compute  $\theta(k)$ , and consequently  $\bar{r}(k)$  and  $u_{\text{ff}}(k)$  using Definition~5.  
 (d) set  $k \rightarrow k+1$  and repeat step (2a)–(2c).

*Remark 6.* The bootstrapping step can also be performed in a batch-wise approach instead of at each sample. By starting to bootstrap with  $k > k_s$  and  $k_s > 1$ , the estimate  $\theta(k)$  has already converged to avoid divergence at the start of the algorithm.

**4 | DESIGN ASPECTS**

In this section, implementation aspects are provided, including a stable approximation of the adaptive filters  $T_{\text{ur}}$ ,  $T_{\text{yr}}$ , and  $\hat{F}^{\text{opt}}$ , the inclusion of an exponential forgetting factor in the Definition 5, and how to choose the optimal filter  $L$  in Figure 4.

**4.1 | Stable approximations for on-line bootstrapping**

The filters  $\hat{T}_{\text{yr}}(\theta_{\text{bs}})$ ,  $T_{\text{ur}}(\theta_{\text{bs}})$  and  $\hat{F}^{\text{opt}}(\theta_{\text{bs}})$  can be unstable for some  $\theta_{\text{bs}}(k)$  such that  $\hat{u}_r$  and  $\hat{y}_r$  can become unbounded which is undesirable. The filters defined in (20) and (28) are expected to be stable by design. Therefore, the following procedure presents a stable approximation approach where the possibly unstable poles in  $\hat{T}_{\text{yr}}(\theta_{\text{bs}})$ ,  $T_{\text{ur}}(\theta_{\text{bs}})$  and  $\hat{F}^{\text{opt}}(\theta_{\text{bs}})$  are approximated with stable poles.

Procedure 2 creates a stable approximation that has zero magnitude error, similar to zero-magnitude-error-tracking-control (ZMETC) to obtain a stable approximation in Reference 29, see Figure 5. Alternative approaches are conceptually possible, for example, ZPETC<sup>30</sup> to obtain a zero-phase error estimate if desired, by replacing unstable poles by non-minimum phase zeros. The motivation for Procedure 2 is that it replaces unstable poles by stable poles which is expected since the filters  $\hat{F}(\theta_{\text{bs}})$ ,  $\hat{T}_{\text{ur}}(\theta_{\text{bs}})$ , and  $\hat{T}_{\text{yr}}(\theta_{\text{bs}})$  are expected to be stable as well.

**Procedure 2.** Zero magnitude stable approximation

1. Separate the denominator of  $G$  into a stable part  $A_s$  and an unstable part  $A_u$ , that is,

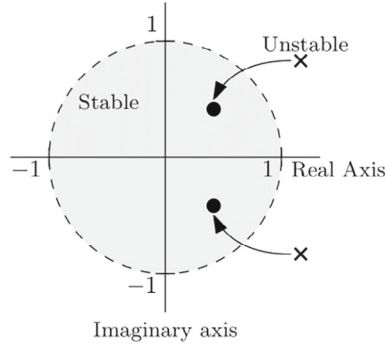
$$G(q^{-1}) = \frac{B(q^{-1})}{A_u(q^{-1})A_s(q^{-1})}, \quad (33)$$

where  $A_u(q^{-1}) = a_0 + a_1q^{-1} + \dots + a_nq^{-n}$ .

2. The stable approximation of  $G$  such that  $|G| = |\hat{G}_s|$  is given by

$$\hat{G}_s(q^{-1}) = \frac{B(q^{-1})}{A_u^f(q^{-1})A_s(q^{-1})}, \quad (34)$$

where  $A_u^f(q^{-1}) = a_n + a_{n-1}q^{-1} + \dots + a_0q^{-n}$ .



**FIGURE 5** Graphical illustration of the stable approximation in Procedure 2 where unstable poles (x) are approximated with stable poles (●) with zero magnitude error.

*Remark 7.* If the  $P$  contains a rigid-body mode, i.e., poles at  $1 + 0i$ , then  $F^{\text{opt}}(\theta_0)$  is unstable. To avoid this for  $\hat{F}(\theta_{\text{bs}})$  the basis functions can be altered such that the poles are shifted slightly into the unit disk.

## 4.2 | Filtering of $\theta(k)$ for bootstrapping

The parameter estimate  $\theta$  obtained with Procedure 1 can be erratic when few data points are available, potentially leading to divergence of the bootstrapping approach. Directly using these erratic parameters for bootstrapping and feedforward is undesirable. Therefore, the filter  $L$  in Figure 4 is introduced to filter the output of the recursive IV algorithm before bootstrapping, for example, with a low-pass filter characteristic. In this paper, a moving average filter is used such that

$$\theta_{\text{bs}}(k) = \underbrace{\frac{(1 - \gamma)q^{-1}}{1 - \gamma q^{-1}}}_{L(q^{-1})} \theta(k), \quad (35)$$

with  $\gamma = e^{-2\pi f_c T_s}$  where  $f_c \geq 0$  is the cut-off frequency and  $T_s$  the sample time. Note that  $f_c$  is a tuning parameter, decreasing  $f_c$  yields that  $\theta_{\text{bs}}$  is smoother, and for  $\lim_{f_c \rightarrow \infty} L(q^{-1}, f_c) = q^{-1}$ , i.e., no filtering. Note that with  $f_c > 0$  the filter  $L$  introduces a modeling error in  $\hat{F}^{\text{opt}}$ ,  $\hat{T}_{ur}$  and  $\hat{T}_{yr}$ . Hence, there is a trade-off between performance and robustness in the tuning of  $f_c$  that is preferably large.

## 4.3 | Incorporating system knowledge

Modern control systems are implemented in discrete-time, i.e., a part of the plant that is introduced through zero-order-hold discretization and system delays are often known in advance. To use this knowledge instead of estimating it, the system can be parameterized as

$$P(q^{-1}, \theta_0) = \tilde{P}(q^{-1}) \frac{B_0(q^{-1}, \theta_0)}{A_0(q^{-1}, \theta_0)}, \quad (36)$$

where  $\tilde{P}(q^{-1})$  contains the known part. Consequently, by using  $\tilde{u}(k) = \tilde{P}(q^{-1})u(k)$  instead of  $u(k)$  in Definition 5, the adaptation law is adapted accordingly. Moreover, the implementation of the adaptive feedforward becomes  $\tilde{C}_r(q^{-1}) = C_r(q^{-1})\tilde{P}(q^{-1})$  accordingly.

*Remark 8.* Alternatively, the inverse of  $\tilde{P}$  can be implemented in the feedforward controller, i.e.,  $\tilde{C}_{\text{ff}} = \tilde{P}^{-1}C_r$ , if  $\tilde{P}^{-1}$  is stable. Moreover, a combination where the numerator and denominator of  $\tilde{P}$  are implemented in  $C_r$  and  $C_{\text{ff}}$  respectively is also possible.

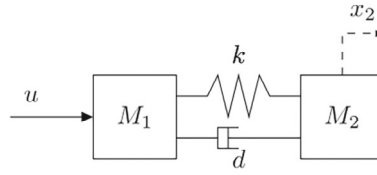


FIGURE 6 Case study: Simplified model of the short-stroke wafer stage dynamics in the x-direction.

## 5 | WAFER-STAGE CASE STUDY

In this section, a case study is performed that mimics the model and the type of references that are encountered in the wafer-stage motivating example in Figure 2. A simplified model of the short-stroke stage dynamics in the x-direction is depicted in Figure 6, that is, it is modeled as two masses with a flexible element in-between.

### 5.1 | Wafer-stage model

The model is given by

$$\begin{aligned} P(q^{-1}) &= \tilde{P}(q^{-1}) \frac{\Psi_B(q^{-1})\theta_0^B}{\Psi_A(q^{-1})\theta_0^A} \\ &= \tilde{P}(q^{-1}) \frac{9.26 \cdot 10^{-6}(1 + q^{-1})}{(1 - q^{-1})^2(1 - 1.967q^{-1} + 0.9977q^{-2})}, \end{aligned}$$

where  $\tilde{P}$  contains a discrete-time approximation of 2.2 samples delay, and the basis functions are

$$\Psi_B(q^{-1}) = \left( \frac{1 - q^{-1}}{T_s} \right) \quad (37)$$

$$\Psi_A(q^{-1}) = \left[ \left( \frac{1 - q^{-1}}{T_s} \right)^2 \quad \left( \frac{1 - q^{-1}}{T_s} \right)^3 \quad \left( \frac{1 - q^{-1}}{T_s} \right)^4 \right]^T \quad (38)$$

with true system parameters  $\theta_0^A = [39.4 \quad -0.018 \quad 0.00031] \cdot 10^{-5}$  and  $\theta_0^B = -24.4 \cdot 10^{-5}$ . A Bode plot of the system is shown in Figure 7 (—). A stabilizing feedback controller is designed given by

$$C_{fb}(q^{-1}) = \frac{0.69 + 0.006q^{-1} - 0.68q^{-2}}{1 - 1.672q^{-1} + 0.696q^{-2}}, \quad (39)$$

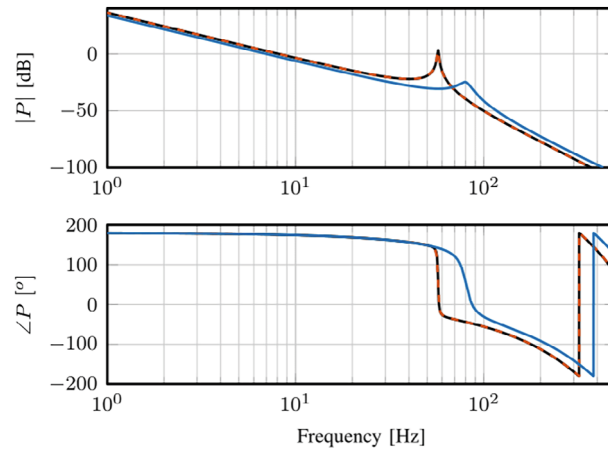
this results in a bandwidth of approximately 10 Hz. Moreover, the system output is subject to additive normally distributed zero-mean noise with variance  $\sigma_v = 10^{-4}$ .

The aim is to track a point-to-point reference with a random step size  $\Delta r_i$  and random dwell times  $\Delta t_i$ , similar to the actual references encountered in the wafer stage setup. Several point-to-point reference tasks are depicted in Figure 8 which are shifted to the origin for comparison.

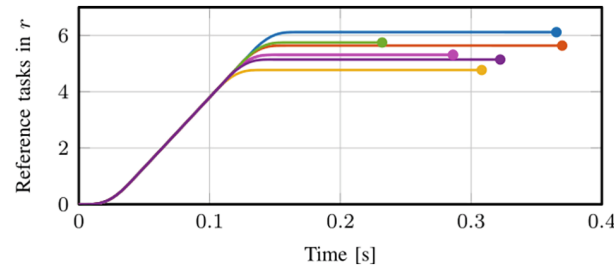
### 5.2 | Tuning for IV-based feedforward control

In view of Assumption 2, the basis functions for feedforward are selected as in (38). The initial parameter estimates are chosen as

$$\begin{aligned} \theta^A(0) &= \begin{bmatrix} 0.75 & 1.25 & 1.5 \end{bmatrix} \theta_0^A, \\ \theta^B(0) &= 0, \end{aligned}$$



**FIGURE 7** Case study: Bode plot of the true system (—) and the model that is obtained with recursive approximate optimal IV identification with on-line bootstrapping (---) with the initial estimate (—).



**FIGURE 8** Case study: Individual reference tasks taken from the point-to-point reference in the simulation case study. Note that each tasks has a different random step size and dwell time, the end of a task is indicated with the markers (●).

such that there is a significant initial error. Moreover,

$$P(0) = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0.2 \end{bmatrix} \cdot 10^4,$$

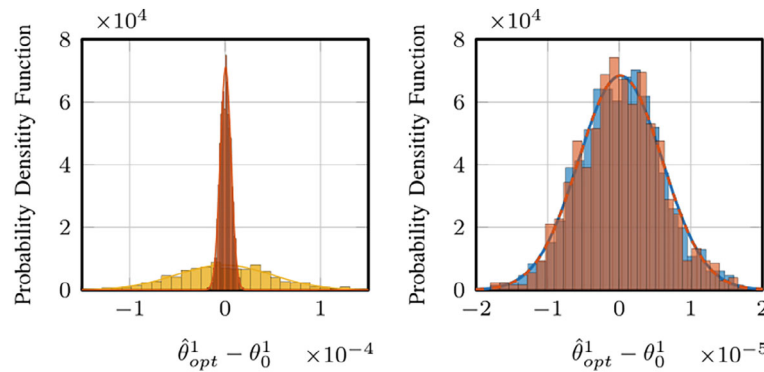
and the cut-off frequency in (35)  $f_c = 5$  Hz to filter out the high-frequency variations in  $\theta$  before on-line bootstrapping.

### 5.3 | Result: Estimation performance through Monte Carlo simulation

To investigate the convergence and the variance of the recursive IV estimator with on-line bootstrapping, a set of 1000 Monte-Carlo simulations is performed with random noise realizations where each simulation consists of 60 point-to-point tasks as in Figure 1. The optimal recursive IV estimator with the true system parameters is used in the simulations to compare the theoretical lower bound for the variance with the obtained approximate optimal IV variance. In addition, a simulation is performed without bootstrapping such that C1 and C2 in Theorem 1 are satisfied and the estimator is asymptotically unbiased, but the variance is not optimized. The converged estimation error  $\theta - \theta_0$  of all simulations is shown in the histogram plot in Figure 9 for the first parameter with Procedure 1 (left) and with optimal IV (right).

From the results, it can be concluded that;

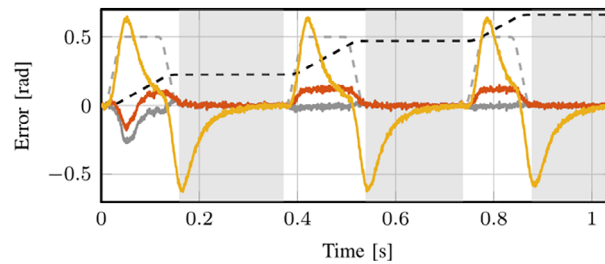
1. the estimation error is normally distributed as shown in Theorem 1,



**FIGURE 9** Case study: Histogram plot of the estimation error for  $\theta_2$  over 1000 Monte Carlo simulations for optimal IV (■) and with a fixed set of instrumental variables (■), that is, without bootstrapping (left). By employing the presented bootstrapping method in Procedure 1 the estimator variance (■) (right) is significantly improved and closely approximates the optimal variance (■).

**TABLE 1** Mean and standard deviation of the converged error with 1000 Monte Carlo simulations for optimal IV, approximate optimal IV with bootstrapping and IV without bootstrapping.

	Opt. IV	IV + bootstrapping	IV
Mean	$1.61 \times 10^{-7}$	$1.59 \times 10^{-7}$	$-1.17 \times 10^{-6}$
Standard deviation	$5.28 \times 10^{-6}$	$5.53 \times 10^{-6}$	$5.06 \times 10^{-5}$



**FIGURE 10** Case study single simulation: Convergence of the tracking error  $e_r$  in (2) (—) with on-line IV-based feedforward in Procedure 1 in comparison to the tracking error  $e_r$  without feedforward (—). A scaled version of the reference (—) and its velocity (—) are also shown, the areas (■) indicate  $\Delta T$  where performance is required, that is, where  $e$  (—) in (9) is equal to  $e_r$ .

- the variance of the on-line bootstrapping IV estimator (■) closely resembles the optimal IV estimator variance (■), and
- the IV estimator without bootstrapping (■) has much larger variance as expected.

To summarize the results, the mean and variance of the optimal estimator and the approximate bootstrapping IV estimator are given in Table 1, confirming these observations.

The obtained plant estimate  $C_r C_{ff}^{-1} \approx P$ , as in (10), is shown in Figure 7 together with the true plant  $P_0$  and the initial estimate. This result shows that indeed the true system is accurately recovered using the adaptive estimator.

## 5.4 | Result: Feedforward performance

The results of using the estimate  $\theta(k)$  obtained with approximate optimal on-line bootstrapping IV for feedforward control, as in Figure 3, is shown in Figure 10. The tracking error  $e = r - y$  is shown in addition to the feedback error  $e_r = \bar{r} - y$ . Moreover, the tracking error  $e_r$  is also shown without feedforward control for comparison.



The results show that;

1. with feedforward a significant performance improvement is obtained within the first task w.r.t. only feedback control.
2. in the part where the velocity of the reference is zero, where performance is crucial in the wafer-stage application, both error  $e_r$  and  $e$  are identical because of the underlying Assumption 1, indicating that good performance is obtained by properly estimating the system dynamics from data while operating in closed-loop.

This shows that indeed for varying point-to-point motion tasks good estimation and tracking performance can be obtained with IV-based adaptive feedforward control by using on-line bootstrapping.

## 6 | EXPERIMENTS ON A BENCHMARK MOTION SYSTEM

A benchmark motion system is used for experimental validation in a practical situation, i.e., where the system is not fully captured by the basis functions. The introduced approximate optimal IV approach with on-line bootstrapping is used for feedforward, moreover, a comparison is made with a recursive least squares (RLS) approach using the same basis functions. For a comparison with existing batch-wise approaches, see, for example, Reference 31.

### 6.1 | Experimental setup and tuning

The experimental setup is shown in Figure 11 which consists of two masses that are connected by a flexible axle, as schematically depicted in Figure 6. The position of the collocated mass is used for feedback which is measured by an encoder with a resolution of  $\frac{2\pi}{2000}$  rad, this induces additional quantization noise.

A frequency response function of the setup is measured and depicted in Figure 12 in (—), which contains similar dynamics as the model used in Section 5. Therefore, the same set of basis functions is used. The setup also contains 2.2 samples of input-output delay and a sampling zero that is included in  $\tilde{P}$  as in (36), that is, this is not estimated. For convenience, the other settings and initial parameters are identical so the simulation case study and provided in Section 5.2.

### 6.2 | Experimental results

The resulting system estimates  $C_r C_{ff}^{-1}$  and errors obtained in the experiments are shown in Figures 12 and 13 respectively, from which the following conclusions can be made.

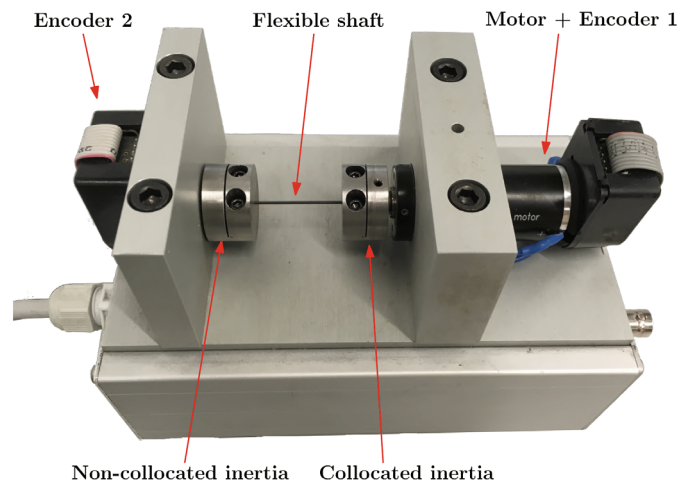
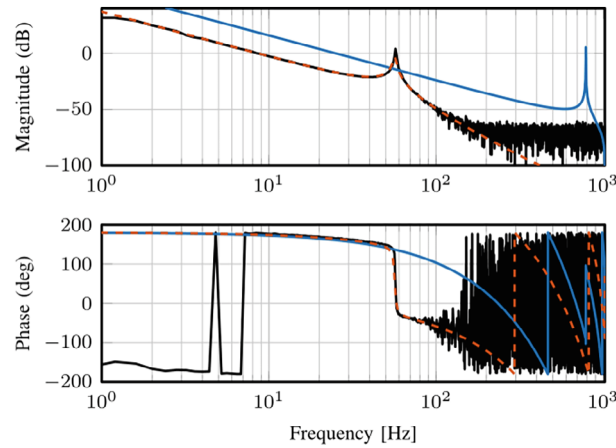
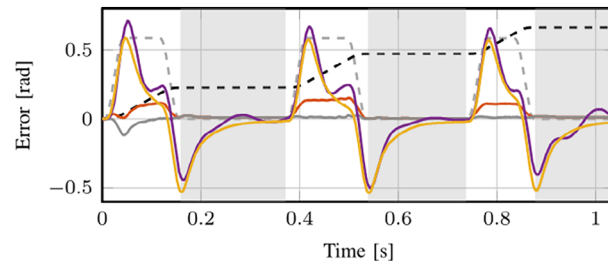


FIGURE 11 Benchmark motion system used for experimental validation.



**FIGURE 12** Experimental validation: Bode plot of the measured frequency response function (FRF) of the experimental setup (—), the plant model  $C_r C_{\pi}^{-1}$  obtained with approximate optimal on-line IV feedforward (---) and with recursive least squares identification (—).



**FIGURE 13** Experimental validation: Tracking error  $e_r$  (—) in (2) obtained with Procedure 1 yielding good performance in comparison with the case without feedforward (—). Also a recursive least-squares based feedforward (—) is used that performs poor due to estimation bias. A scaled version of the reference (---) and its velocity (---) are also shown, and the areas (■) indicate where performance is required and where  $e(9)$  is equal to  $e_r$ .

1. The obtained plant estimate for feedforward, with the converged parameters, corresponds well with the measured FRF, indicating that a good model is obtained. In contrast to the recursive LS estimator, which yields a very poor estimate of the system due to an estimation bias.<sup>13</sup>
2. The tracking error with IV-based on-line feedforward is significantly smaller compared with only feedback control, that is, within 0.2 s the error converges to a small value and remains small. This indicates that the feedforward parameters converge fast to the actual system values.
3. The tracking performance with Recursive LS-based on-line feedforward is even worse than without feedforward due to the biased estimates that are obtained.

These results confirm that in the presence of noise the recursive IV-based estimator is able to obtain a good system estimate and consequently good performance in the dwell time as required for the actual wafer-stage application.

## 7 | CONCLUSIONS

On-line learning of adaptive feedforward parameters for the tracking of varying point-to-point references is enabled through an approximate optimal instrumental variable (IV) estimator with an on-line bootstrapping approach. Traditional batch-wise learning approaches, such as ILC, are not applicable due to the non-resetting behavior of such references. In this paper, feedforward parameters are estimated in an on-line setting on the basis of data, that is, the reference and the system input and output are used. This estimator yields an unbiased estimate with approximate optimal variance in a

closed-loop setting with measurement noise. A simulation case study of a wafer-stage setup and experiments on a benchmark motion system shows the immediate performance improvement. Ongoing work focuses on the implementation for systems that have slowly varying parameters by incorporating a forgetting factor in the on-line IV algorithm, such that the feedforward parameters are updated accordingly.

## ACKNOWLEDGMENTS

The authors want to thank Thijs Sieswerda and Gijs van der Veen for their contributions to this work. The research leading to these results has received funding from the European Union H2020 program under grant agreement n. 637095 (Four-By-Three) and ECSEL-2016-1 under grant n. 737453 (I-MECH), and the ECSEL Joint Undertaking under grant agreement n. 101007311 (IMOCO4.E).

## CONFLICT OF INTEREST

The authors declare no potential conflict of interests.

## DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

## ORCID

Noud Mooren  <https://orcid.org/0000-0002-9725-5389>

## REFERENCES

- van der Veen G, Stokkermans J, Mooren N, Oomen T. How learning control improves product quality in semiconductor assembly equipment. Paper presented at: ASPE 2020 Spring-Design and Control of Precision Mechatronic Systems. 2020 1–5.
- Boeren F, Bareja A, Kok T, Oomen T. Frequency-domain ILC approach for repeating and varying tasks: with application to semiconductor bonding equipment. *IEEE/ASME Trans Mechatron*. 2016;21(6):2716–2727.
- Boeren F, Bruijnen D, van Dijk N, Oomen T. Joint input shaping and feedforward for point-to-point motion: automated tuning for an industrial nanopositioning system. *Mechatronics*. 2014;24(6):572–581.
- Bristow DA, Tharayil M, Alleyne AG. A survey of iterative learning control. *IEEE Control Syst Mag*. 2006;26(3):96–114.
- Oomen T. Control for precision mechatronics. *Encyclopedia of Systems and Control*. Springer; 2019.
- Strijbosch N, Tiels K, Oomen T. Hysteresis feedforward compensation: a direct tuning approach using hybrid-mem-elements. *IEEE Control Syst Lett*. 2021;6:1070–1075.
- Veronesi M, Visioli A. Automatic tuning of feedforward controllers for disturbance rejection. *Ind Eng Chem Res*. 2014;53(7):2764–2770.
- Altn B, Willems J, Oomen T, Barton K. Iterative learning control of iteration-varying systems via robust update laws with experimental implementation. *Control Eng Pract*. 2017;62:36–45.
- Bolder J, Oomen T. Rational basis functions in iterative learning control—with experimental verification on a motion system. *IEEE Trans Control Syst Technol*. 2015;23(2):722–729.
- van de Wijdeven J, Bosgra O. Using basis functions in iterative learning control: analysis and design theory. *Int J Control*. 2010;83(4):661–675.
- Gorinevsky D, Torfs DE, Goldenberg AA. Learning approximation of feedforward control dependence on the task parameters with application to direct-drive manipulator tracking. *IEEE Trans Robot Autom*. 1997;13(4):567–581.
- Butler H. Adaptive feedforward for a wafer stage in a lithographic tool. *IEEE Trans Control Syst Technol*. 2012;21(3):875–881.
- Mooren N, Witvoet G, Oomen T. Feedforward motion control: from batch-to-batch learning to online parameter estimation. Paper presented at: 2019 American Control Conference (ACC), IEEE. 2019 947–952.
- Söderström T, Stoica P. *System Identification*. Prentice Hall; 1989.
- Ljung L. *System Identification: Theory for the User, Ser.* Prentice Hall information and system sciences series. Prentice Hall PTR; 1999.
- Gilson M, Garnier H, Young PC, Van den Hof PM. Optimal instrumental variable method for closed-loop identification. *IET Control Theory Appl*. 2011;5(10):1147–1154.
- Janot A, Vandanjon P-O, Gautier M. A generic instrumental variable approach for industrial robot identification. *IEEE Trans Control Syst Technol*. 2013;22(1):132–145.
- Gilson M, Van Den Hof P. Instrumental variable methods for closed-loop system identification. *Automatica*. 2005;41(2):241–249.
- Song F, Liu Y, Xu J-X, Yang X, Zhu Q. Data-driven iterative feedforward tuning for a wafer stage: a high-order approach based on instrumental variables. *IEEE Trans Ind Electron*. 2018;4:3106–3116.
- Fu X, Yang X, Zanchetta P, Tang M, Liu Y, Chen Z. An adaptive data-driven iterative feedforward tuning approach based on fast recursive algorithm: with application to a linear motor. *IEEE Transactions on Industrial Informatics*. IEEE; 2022.
- Boeren F, Blanken L, Bruijnen D, Oomen T. Optimal estimation of rational feedforward control via instrumental variables: with application to a wafer stage. *Asian J Control*. 2018;20(1):1–18.
- Freedman DA. *Statistical Models: Theory and Practice*. Cambridge University Press; 2009.

23. Tjarnstrom F, Ljung L. Using the bootstrap to estimate the variance in the case of undermodeling. *IEEE Trans Automat Contr*. 2002;47(2):395-398.
24. Fujimoto H, Yao B. Multirate adaptive robust control for discrete-time non-minimum phase systems and application to linear motors. *IEEE/ASME Trans Mechatron*. 2005;10(4):371-377.
25. Boeren F, Bruijnen D, Oomen T. Enhancing feedforward controller tuning via instrumental variables: with application to nanopositioning. *Int J Control*. 2017;90(4):746-764.
26. Åström KJ, Wittenmark B. *Adaptive Control*. Second ed. Dover Publications, Inc; 2013.
27. Ioannou PA, Sun J. *Robust Adaptive Control*. Vol 1. PTR Prentice-Hall; 1996.
28. Boeren F, Oomen T, Steinbuch M. Iterative motion feedforward tuning: a data-driven approach based on instrumental variable identification. *Control Eng Pract*. 2015;37:11-19.
29. Butterworth JA, Pao LY, Abramovitch DY. The effect of nonminimum-phase zero locations on the performance of feedforward model-inverse control techniques in discrete-time systems. Paper presented at: 2008 American Control Conference, IEEE. 2008 2696-2702.
30. Tomizuka M. Zero phase error tracking algorithm for digital control. *J Dyn Syst Meas Control*. 1987;109(1):65-68.
31. Mooren N, Witvoet G, Oomen T. From batch-to-batch to online learning control: experimental motion control case study. *IFAC-PapersOnLine*. 2019;52(15):406-411.
32. Söderström T, Stoica P. Instrumental variable methods for system identification. *Circuits Syst Signal Process*. 2002;21(1):1-9.

**How to cite this article:** Mooren N, Witvoet G, Oomen T. On-line instrumental variable-based feedforward tuning for non-resetting motion tasks. *Int J Robust Nonlinear Control*. 2023;33(18):11000-11018. doi: 10.1002/rnc.6925

## APPENDIX

*Proof of Theorem 1.* The proof of Theorem 1 consists of three parts, (i) it is shown that the minimizer of (12) has an analytic expression  $\theta(k)$ , (ii) it is shown that for  $k \rightarrow \infty$  this solution converges to  $\theta_0$  under the conditions in Theorem 1, and (iii) it is shown that the obtained estimator variance, with  $z^{\text{opt}}(k)$  and  $F^{\text{opt}}$ , is equal to the theoretical lower bound.

*Part (i)* The cost function  $V_{\text{IV}}$  is quadratic  $\|x\|^2 = x^\top x$  and linear in  $\theta$ , hence, a sufficient condition for the minimizer is that  $\frac{\partial V(\theta)}{\partial \theta} = 0$ . By substituting (14) in (13) and equating the derivative to zero, it follows that the minimizer is  $\theta(k) = (R_k^\top R_k)^{-1} R_k^\top U_k$  with

$$R_k = \sum_{i=1}^k z(i)F(q^{-1})\phi^\top(i), \quad U_k = \sum_{i=1}^k z(i)F(q^{-1})u(i),$$

and  $(R_k^\top R_k)$  is non-singular.

*Part (ii)* To prove consistency, that is,  $(\theta - \theta_0) \rightarrow 0$  for  $k \rightarrow \infty$  with probability one, under the conditions in Theorem 1, use the relation  $u(k) = \phi^\top \theta_0 + \bar{v}(k)$  with  $\bar{v}(k) = -A_0 v(k)$ , and define  $V_k = \sum_{i=1}^k z(i)F(q^{-1})\bar{v}(i)$  to write  $U_k = R_k \theta_0 + V_k$ . Consequently, the estimation error is

$$(\theta(k) - \theta_0) = (R_k^\top R_k)^{-1} R_k^\top V_k,$$

using Lemma B.2 in Reference 14, where it is shown that under mild conditions

$$\lim_{k \rightarrow \infty} R_k = \mathbb{E}[z(k)F(q^{-1})\phi_r(k)] =: R \quad (\text{A1})$$

$$\lim_{k \rightarrow \infty} V_k = \mathbb{E}[z(k)F(q^{-1})\bar{v}(k)] =: V, \quad (\text{A2})$$

with  $\phi_r(k)$  the noise-free part of  $\phi(k)$ . Consequently, the asymptotic estimation error, i.e., for  $k \rightarrow \infty$ ,  $(\theta_0 - \theta) = (R^\top R)^{-1} R^\top V$  is zero if  $R$  is non-singular, and  $\mathbb{E}z(i)F(q^{-1})\bar{v}(i) = 0$ , which is the case since  $z^{\text{opt}}(k)$  in (16) is uncorrelated with  $v$ .

Part (iii) The final part of the proof is based on [Chapter 8.2],<sup>14</sup> where it is shown that  $(\theta(k) - \theta_0) \rightarrow \mathcal{N}(0, P_{IV})$  for  $k \rightarrow \infty$  with

$$P_{IV} = (R^T R)^{-1} R^T \Gamma R (R^T R)^{-1}, \quad (\text{A3})$$

where  $R = \lim_{k \rightarrow \infty} \sum_{i=1}^k z(i) F(q^{-1}) \phi^T(i)$  under mild assumptions in Reference 32, and

$$\Gamma = \sigma_v^2 \mathbb{E}[F(q^{-1}) H(q^{-1}) z(k) [F(q^{-1}) H(q^{-1}) z(k)]^T], \quad (\text{A4})$$

with  $H(q^{-1}) = A(q^{-1}, \theta_0)$ , a proof of this result can be found in Appendix A8.1 in Reference 14. In Reference 14 it is also proven that

$$P_{IV} \geq P_{IV}^{\text{opt}} = [\sigma_v^2 \mathbb{E}([A_0^{-1} \phi_r(k)]^T [A_0^{-1} \phi_r(k)])]^{-1}. \quad (\text{A5})$$

Finally, it remains to show that by substitution of (16) and (17) in (A3) it follows that  $P_{IV} = P_{IV}^{\text{opt}}$ , which is outlined in detail in Reference 14 [p. 274], which completes the proof. ■