

From Batch-to-Batch to Online Learning Control: Experimental Motion Control Case Study

Noud Mooren* Gert Witvoet*,** Tom Oomen *

* *Control Systems Technology, Department of Mechanical Engineering,
Eindhoven University of Technology, Eindhoven, The Netherlands
(n.f.m.mooren@tue.nl, g.witvoet@tue.nl, t.a.e.oomen@tue.nl)*

** *TNO Technical Sciences, Optomechatronics department, Delft, The
Netherlands.*

Abstract: Data-driven feedforward control can significantly improve the positioning performance of motion systems. The aim of this paper is to exploit the concept of batch-to-batch learning control with basis function, applied in an online fashion. This enables learning within a task while maintaining task flexibility. A recursive least squares optimization is proposed on the basis of input/output data to compute the optimal feedforward parameters. The proposed method is successfully validated in simulation, and applied to a benchmark motion system leading to a major performance improvement compared to only feedback control.

Keywords: Feedforward Control, Learning Control, Parameter Estimation

1. INTRODUCTION

Learning from data can substantially increase control performance in motion systems, e.g., ranging from printing systems (Bolder et al., 2017) and lithography (Blanken et al., 2017) to semiconductor wire-bonding equipment (Boeren et al., 2016). By learning a feedforward signal in the iteration domain, i.e., based on previous motion tasks, the repeating contributions of the error are compensated. These batch-to-batch learning approaches are well developed leading to performance improvement for systems that perform repeating motion tasks (Bristow et al., 2006; Gao and Mishra, 2014). However, standard learning algorithms such as iterative learning control (ILC) cannot cope with varying references hampering industrial deployment.

Recently, learning control algorithms are extended with basis function, enabling learning for system with non-identical tasks (Hoelzle et al., 2011; van de Wijdeven and Bosgra, 2010). In these approaches, the feedforward controller is parameterized using basis function and, instead of learning a signal, the feedforward parameters are learned allowing extrapolation to varying references. The aim is to compute a feedforward controller that reflects the inverse system dynamics resulting in ideal tracking control (Boerlage et al., 2003; Widrow, 1987). An important factor is the specific choice of a basis, representing the plant inverse. In van der Meulen et al. (2008) polynomial basis functions are exploited which leads to a linear optimization problem, however, this only captures systems with unit numerator. In Blanken et al. (2017) rational

basis functions are considered, capturing a wider range of plants. However, this requires a non-linear optimization problem. In Boeren et al. (2014) an alternative approach is presented in which a novel combination of input-shaping and feedforward control is used, allowing flexible batch-to-batch learning with a rational basis, while maintaining a linear optimization problem.

Although batch-to-batch learning allows for substantial performance improvements after each task, learning within a task, i.e., current iteration learning, lacks a fundamental basis. Promising results towards adaptive feedforward control are presented in Butler (2012), however, in such current iteration or adaptive approaches biased estimates can be obtained due to closed-loop issues (van den Hof and Schrama, 1994; Boeren et al., 2015). More specific, in Mooren et al. (2019), a statistical analyses of current iteration learning is presented showing that biased estimates are obtained if noise is present.

The aim of this work is to present an on-line optimization approach enabling fast learning combined with task flexibility. To this extend, the approach in Mooren et al. (2019) is further developed focusing on the implementation aspects, the effects of noise leading to biased estimates are reduced and an experimental motion control case study is performed. The sub-contributions are as follows:

- C_1 the approach in Mooren et al. (2019) is extended for online optimization of feedforward parameters from measurement data,
- C_2 a simulation study is performed for validation,
- C_3 an experimental case study is performed to show the potential of current iteration learning, and confirming the theoretical conclusions in Mooren et al. (2019).

This paper is organized as follows. In Section 2, the feedforward control problem is defined and a feedforward

* The research leading to these results has received funding from the European Union H2020 program under grant agreement n. 637095 (FourByThree) and ECSEL-2016-1 under grant agreement n. 737453 (I-MECH). Also, this work is part of the research programme VIDI with project number 15698, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO).

controller parameterization is proposed. In Section 3, an optimization algorithm is derived to compute the feedforward parameters based on input and output data. In Section 4, a simulation study is performed to indicate the benefit of this method. In Section 5, an experimental case study is performed on a benchmark motion system. Finally, conclusions and ongoing research are presented in Section 6.

2. PROBLEM DEFINITION

In this section, the control setup is introduced and the feedforward control problem is defined in more detail. Furthermore, a rational feedforward parameterization is proposed that allows to optimize feedforward parameters as described in Section 3.

2.1 Control setup and feedforward control problem

Consider the control setup in Fig. 1, including both feedforward and feedback control. Here $C_r \in \mathbb{R}[q^{-1}]$ has the role of an input shaper, $C_{ff}(\theta) \in \mathbb{R}[q^{-1}]$ is the feedforward controller parameterized as function of $\theta \in \mathbb{R}^{n_\theta}$ and C_{fb} is a stabilizing and fixed feedback controller. Furthermore, $P_0 \in \mathcal{R}$ denotes the true system given by,

$$P_0 = \frac{B_0(q^{-1}, \theta_0^b)}{A_0(q^{-1}, \theta_0^a)} \quad (1)$$

in which $B_0(q^{-1})$ and $A_0(q^{-1})$ are the system numerator and denominator polynomials as function of the true system parameters $\theta_0 = [\theta_0^a \ \theta_0^b]$. The notation \mathcal{R} refers to real rational functions and $\mathbb{R}[q^{-1}]$ refers to real polynomial functions in the delay operator q^{-1} .

The aim is to compute the optimal parameters θ on the basis of data $\{u\}$ and $\{y\}$ such that the positioning error is minimized. Therefore, an optimization algorithm is present in Fig. 1, to compute the optimal feedforward parameters. The optimization algorithm is further developed in section 3. Next, consider the following feedforward control goal.

Definition 1. (Feedforward control goal). Compute a causal feedforward controller C_{ff} and input shaper C_r such that the reference induced positioning error

$$e = S(C_r - P_0 C_{ff})r \quad (2)$$

is minimal.

A well-known result in feedforward control is that the feedforward controller must reflect the inverse of the system dynamics to eliminate reference induced errors. This becomes evident by substituting the feedforward controller and input shaper as,

$$C_r C_{ff}^{-1}(\theta) = P_0 \quad (3)$$

in (2) leading to $e(t) \rightarrow 0$ for all time, if the numerator and denominator of P_0 are described by C_r and C_{ff} respectively (Boeren et al., 2014). This result implies that instead of optimizing the positioning error directly, one can alternatively create an estimate of P_0 that satisfies (3) to minimize the reference induced positioning errors (Boeren et al., 2015, 2014; Ho and Enqvist, 2018).

Next, a feedforward controller parameterization is proposed, that is capable of capturing the system dynamics.

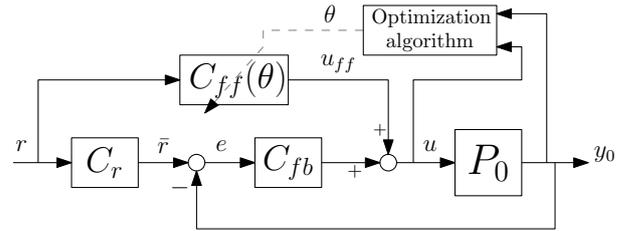


Fig. 1. Control setup with plant P_0 , feedback controller C_{fb} , feedforward controller C_{ff} and input shaper C_r .

2.2 Feedforward controller parameterization

To satisfy (3), the feedforward controller must be parameterized such that it can reflect the system P_0 . this is denoted as $P_0 \in \mathcal{C}$, where \mathcal{C} is the set of possible feedforward controllers defined as follows.

Definition 2. Consider the feedforward controller parameterization,

$$\mathcal{C} = \left\{ (C_r, C_{ff}(\theta)) \mid \begin{array}{l} C_{ff}(\theta) = A(q^{-1}, \theta) \\ C_r = B_0(q^{-1}) \end{array}, \theta \in \mathbb{R}^{n_\theta} \right\} \quad (4)$$

here,

$$A(q^{-1}, \theta_a) = \sum_{i=1}^{n_a} \psi_i(q^{-1}) \theta_i \quad (5)$$

with parameters

$$\theta = [\theta_1 \ \theta_2 \ \dots \ \theta_{n_a}]^\top \in \mathbb{R}^{n_a}, \quad (6)$$

and basis functions

$$\Psi = [\psi_1 \ \psi_2 \ \dots \ \psi_{n_a}]^\top \in \mathbb{R}[q^{-1}], \quad (7)$$

such that $C_{ff}(q^{-1}, \theta) = \Psi^\top(q^{-1})\theta$.

Furthermore, the input shaper must satisfy that

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

which implies that C_r has unit d.c. gain to avoid scaling of the reference, see Boeren et al. (2014) for further details.

Remark 3. Note that the input shaper C_r is fixed to (a scaled version) of B_0 , which is assumed to be known for the sake of simplicity. Hence, in the remainder of this work θ_0 refers to θ_0^b . The approach presented next can naturally be extended to the general case where C_r is also optimized.

To conclude, the feedforward control problem is defined and a parameterization is proposed. The following section focuses on the optimization of $C_{ff}(\theta)$.

3. FEEDFORWARD PARAMETER OPTIMIZATION

The previous section shows that the ideal feedforward controller is given by (3), from which it follows that the error is minimized if the system P_0 is exactly represented by $\mathcal{C}(\theta)$. The aim of this section is to propose a procedure to compute the feedforward parameters.

3.1 Defining the optimization problem

The optimization problem is stated to compute θ on the bases of input data u and measurement data y_m in a closed-loop setting.

Consider the framework in Fig. 2, the aim is to minimize the quadratic cost function,

$$\theta_{\text{opt}} = \min_{\theta} \frac{1}{2} \sum_{i=0}^k \epsilon^2(i). \quad (9)$$

with objective function $\epsilon(k)$,

$$\epsilon(k, \theta) = \bar{u}(k) - \bar{y}_m(k, \theta) \quad (10)$$

in which $\bar{u}(k)$ and $\bar{y}_m(k, \theta)$ are,

$$\bar{u}(k) = \mathcal{B}(q^{-1})u(k) \quad (11)$$

$$\bar{y}_m(k, \theta) = \mathcal{A}(q^{-1}, \theta)y_m(k) \quad (12)$$

where

$$\mathcal{A}(z, \theta) = C_{\text{ff}}(z, \theta)F(z) \quad (13a)$$

$$\mathcal{B}(z) = C_r(z)F(z) \quad (13b)$$

are filtered versions of the plant numerator in C_r and the basis functions in C_{ff} . Note that this is in principle similar to an equation error method in, e.g., Ljung (1999). Additionally, a stable and causal filter $F(z) \in \mathcal{R}$ is present, which is ideally close to 1, and used to improve robustness against noise as motivated in the remainder of this section.

The measured signal y_m may contain noise v and quantized data. This signal is filtered with the basis functions contained in C_{ff} , which are assumed to be of the form,

$$\psi(z) = \left(\frac{z-1}{T_s z} \right)^n \quad (14)$$

being discrete-time equivalents of the continuous-time derivative operator s^n , and a natural choice to parameterize the feedforward controller, see, e.g., Blanken et al. (2017); Boeren et al. (2015). This implies computation of the n^{th} -order derivative of the potentially noisy signal y_m , leading to a bad signal to noise ratio. Therefore, by choosing the filter $F(z)$ as an n^{th} -order low-pass filter,

$$F(z) = \left(\frac{1 - e^{-2\pi f_c T_s}}{z - e^{-2\pi f_c T_s}} \right)^n \quad (15)$$

with n the largest order of $C_{\text{ff}}(\theta)$ and f_c the cut-off frequency, noise effect are mitigated as shown next. Furthermore, this specific choice of $F(z)$ is motivated by the fact that performance is mainly focusing on the frequency range below the bandwidth, whereas noise and quantization effects become more dominant in the higher frequency range.

Remark 4. Note that if F would only be present in \mathcal{A} and not in \mathcal{B} , the parameterization would no longer span the required solution space. Since the poles of $F(z)$ appear in \mathcal{A} and cannot be placed elsewhere by θ .

Next, it is shown that if noise is not present and $P_0 \in \mathcal{C}$, minimization of (9) leads to minimization of (2), i.e., $\theta_{\text{opt}} \rightarrow \theta_0$. To show this rewrite the objective function as

$$\epsilon(k, \theta) = \mathcal{B}(q)u(k) - \mathcal{A}(q, \theta)y_m(k) \quad (16)$$

substitution of $y_m(k) = P_0 u(k) + v(k)$, with (13) and (1) leads to

$$\epsilon(k, \theta) = F \left(B_0 - \Psi^{\top} \theta \frac{B_0}{A_0(\theta_0)} \right) u(k) - F \Psi^{\top} \theta v(k) \quad (17)$$

where the arguments q^{-1} and k are omitted for the ease of notation. This shows that if $v = 0$, the latter term vanishes, and if $\theta \mapsto \theta_0$ the objective function becomes zero and subsequently the cost function (9) is minimized by satisfying (3). If $v \neq 0$ the filter $F(z)$ acts as an additional

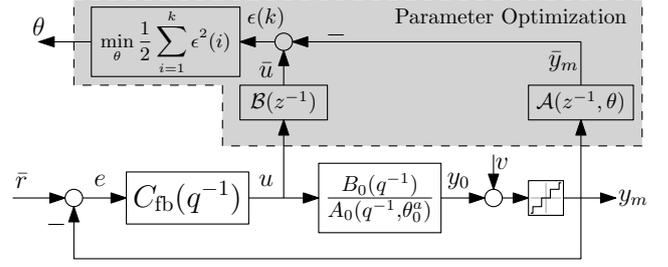


Fig. 2. Parameter optimization setting.

low-pass filter on the noise term, i.e., for high-frequencies noise is suppressed.

To conclude, this analysis implies that biased estimates are obtained if noise is present, and a detailed statistical analysis in Mooren et al. (2019) confirms this for $F = 1$. In this work it is shown that the effect of noise leading to bias can be mitigated by implementing an additional filter $F(z)$ which acts as a low-pass filter.

3.2 Optimization algorithm

In this section, the proposed optimization problem is written into a least-squares problem with an analytic solution. To compute the solution to this problem efficiently for current iteration learning, a procedure is provided based on recursive least squares (RLS).

First define the following

$$\mathcal{E} = \begin{bmatrix} \epsilon(1) \\ \epsilon(2) \\ \vdots \\ \epsilon(k) \end{bmatrix} \quad \Phi = \begin{bmatrix} \phi(1) \\ \phi(2) \\ \vdots \\ \phi(k) \end{bmatrix} \quad \bar{U} = \begin{bmatrix} \bar{u}(1) \\ \bar{u}(2) \\ \vdots \\ \bar{u}(k) \end{bmatrix} \quad (18)$$

where $\phi(k) = F(q^{-1})\Psi^{\top}(q^{-1})y_m(k)$ such that the optimization problem is alternatively written as the least squares problem

$$\bar{U} = \Phi \theta \quad (19)$$

with solution

$$\theta = (\Phi^{\top} \Phi)^{-1} \Phi^{\top} \bar{U}. \quad (20)$$

The feedforward optimization problem can be solved batch-wise by collecting a set of data and solving (20).

For online optimization the solution (20) is not efficiently computed, since it takes all data up to the current iteration into account. Therefore, the problem is solved in a recursive fashion. The solution to (19) at time k is alternatively written as function of current inputs and the previous estimate $\theta(k-1)$,

$$\theta(k) = \theta(k-1) + K(k) (\bar{u}(k) - \phi^{\top} \theta(k-1)) \quad (21)$$

in which the time dependent learning gain $K(k)$ is given by

$$K(k) = P(k) \phi(k) \quad (22)$$

and the matrix $P(k)$ is recursively computed as follows

$$P(k) = P(k-1) [I - \phi(k) \Sigma \phi(k)^{\top} P(k-1)] \quad (23)$$

where

$$\Sigma = (I + \phi(k)^{\top} P(k-1) \phi(k))^{-1} \quad (24)$$

with initial conditions $P(t_0) = (\Phi(t_0)^{\top} \Phi(t_0))^{-1}$ and $\theta(t_0)$. A detailed derivation of the RLS algorithm can be found in (Åström and Wittenmark, 2013, Chapter 2), Goodwin and Sin (2014).

The following procedure is proposed for online optimization of the parameter using the RLS in (21) - (24).

Procedure 5. (Online parameter optimization).

- (1) Define an initial parameter estimate $\theta(t_0)$ and initial condition $P(t_0)$.
- (2) Compute the learning gain $K(k)$ with (22) - (24).
- (3) Compute the parameters $\theta(k)$ using (21).
- (4) Update the controller $C_{ff}(\theta(k))$ using $\theta(k)$ and start at step 2 for the next iteration with $k = k + 1$.

This completes the optimization algorithm and analyses. In the remainder of this work, the proposed method is further elaborated by means of a simulation example and an experimental case study.

4. SIMULATION STUDY

In this section, a simulation study is performed to validate the proposed method in an ideal situation, i.e., without measurement noise and if $P_0 \in \mathcal{C}$. Furthermore, it is shown that the conventional parameterization with $F = 1$ fails if quantization and noise effects are present, whereas, with the proposed low-pass filter the estimate is improved. Finally, it is shown that procedure 5 is capable of optimizing the feedforward parameters in an on-line fashion.

4.1 Simulation model and control goal

Consider a fourth-order system P_0 as depicted in Fig. 3, with 3 samples delay and the denominator is parameterized as (5) where,

$$\theta_0 = [3.7 \cdot 10^{-4}, -1.7 \cdot 10^{-7}, 0.3 \cdot 10^{-8}]^T \quad (25)$$

and basis functions

$$\psi_1(z) = \left(\frac{z-1}{T_s z}\right)^2, \psi_2(z) = \left(\frac{z-1}{T_s z}\right)^3, \psi_3(z) = \left(\frac{z-1}{T_s z}\right)^4$$

resulting in

$$P_0(z) = \frac{4.7496 \cdot 10^{-6}(z+1)^2}{z^3(z-1)^2(z^2-1.969z+0.998)}. \quad (26)$$

This system closely represents the experimental setup that is used in Section 5. Furthermore, a PD-type feedback controller with low-pass filter,

$$C_{fb}(z) = \frac{0.83(z+1)(z-0.9898)}{(z-0.8575)(z-0.8314)} \quad (27)$$

is designed resulting in an open-loop bandwidth of 15 Hz.

Throughout the simulation, the reference is taken as a typical point-to-point motion task. To simulate the effect of an encoder the measured output y_m is quantized with a resolution of $2\pi/2000$ radians which is common in practical applications. Next, the input shaper C_r is parameterized as in definition 2,

$$C_r(z) = \frac{1}{4} \frac{z^2 + 2z + 1}{z^3} \quad (28)$$

which is a scaled version of the plant numerator such that it satisfies (8) with the 3-samples of delay included. The feedforward controller C_{ff} is parameterized as in (4) with basis function ψ_1, ψ_2 and ψ_3 . These basis functions correspond to the second, third and fourth order derivative operators. In terms of feedforward control these can be considered as acceleration, jerk and snap feedforward which are automatically tuned.

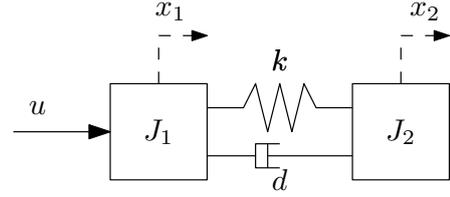


Fig. 3. Two mass-spring-damper system.

4.2 Simulation results

The feedforward parameters are estimated using the proposed approach, i.e., with the RLS algorithm. Multiple simulations with and without quantization noise are performed, to show the effect of the additional filtering with $F(z)$. Finally, the estimation errors are compared.

The following three simulation are performed;

- (1) without quantization noise and $F = 1$, which is considered as the ideal case,
- (2) with quantization noise and $F = 1$,
- (3) with quantization noise and $f_c = 10$ Hz in (15).

The estimation error for the first parameter is depicted on a logarithmic scale in Fig. 4. The first simulation without noise (—) shows that the estimate converges closely to the true parameter up to numerical accuracy. The second simulation with noise (—) clearly shows a major estimation error and slow convergence. This is improved by using the proposed filtering (—), resulting in fast convergence and a small error of $2 \cdot 10^{-8}$.

The obtained plant estimate given by (3) is depicted in Fig. 5 for the latter two simulations. This clearly shows that a poor estimate of the true plant P_0 (—) is obtained if quantization noise is present and $F = 1$ (—). The proposed approach creates a good estimate of the true plant despite the presence of quantization effects (—).

To conclude, these results confirm that the effect of quantization noise is severely reduced by using the filter $F(z)$, leading to accurate and fast estimation of the system dynamics.

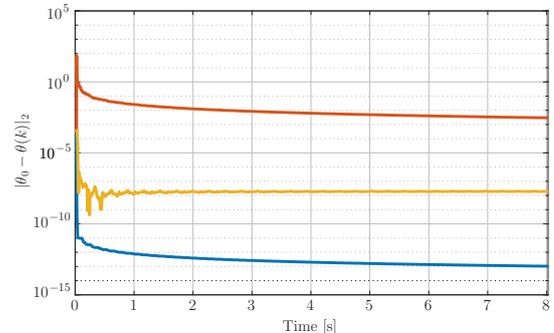


Fig. 4. 2-norm of the difference between the true system parameter θ_0 and (i) estimate ideal simulation (—), (ii) estimate with quantization and $F = 1$ (—) and (iii) estimate obtained with proposed approach with quantization and $f_c = 10$ Hz (—).

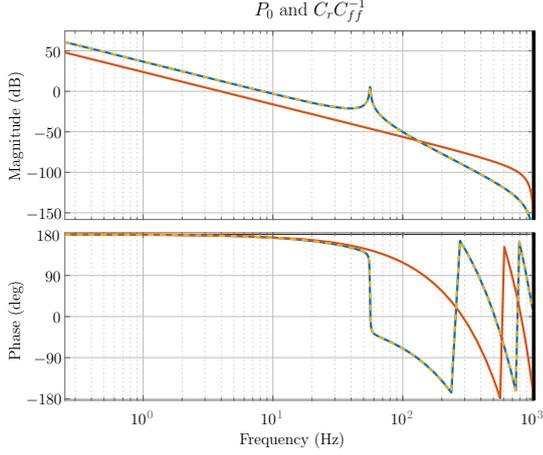


Fig. 5. Plant P_0 (—), obtained estimate with proposed approach and quantization effects (---) and plant estimate with quantization effects and $F = 1$ (—).

5. EXPERIMENTAL VALIDATION

In this section, the proposed feedforward optimization method is applied to a benchmark motion system. In the previous section, the proposed method is validated in a controlled situation, however, during the experiments many undesired effects are present including; sensor quantization, measurement noise, delays and model mismatches $P_0 \notin \mathcal{C}$. The aim of this experimental case study is to show the potential of this method in a practical situation.

5.1 Experimental setup

The experimental setup is depicted in Fig. 6, it consists of two rotating inertias J_1 and J_2 connected with a flexible shaft. The combined inertia is approximately given by $3.68 \cdot 10^{-4} \text{ kg m}^2$. The rotation of both inertias is measured using incremental encoders with a resolution of $2\pi/2000$ radians. The control goal is to minimize the reference induced positioning error (2) of the non-collocated inertia J_2 . Furthermore, the same feedback controller (27) as in the simulations is used for the experiments, again resulting in an open-loop bandwidth of approximately 15 Hz.

5.2 Feedforward controller optimization

A simplified representation of the setup is given in Fig. 3, where the mapping from u to the position of second inertia J_2 is given by the following transfer function.

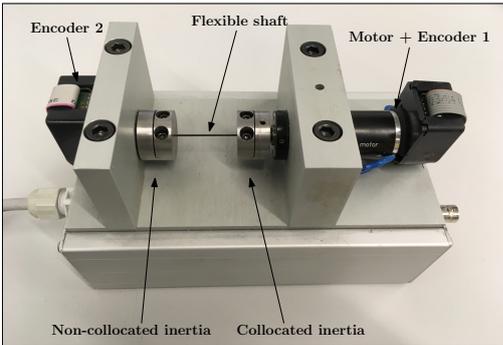


Fig. 6. Experimental setup.

$$P_{\text{ncol}}(s) = \frac{1}{s^2} \frac{ds + k}{J_1 J_2 s^2 + (J_1 + J_2) ds + (J_1 + J_2) k} \quad (29)$$

By assuming that damping is small this becomes

$$P_{\text{ncol}}(s) \approx \frac{1}{\frac{J_1 J_2}{k} s^4 + (J_1 + J_2) s^2}. \quad (30)$$

Hence, to minimize the positioning errors, the feedforward controller C_{ff} should approximate the plant denominator as shown in (3). Therefore, the following two basis functions are considered dominant and used in C_{ff} ,

$$\psi_1(q^{-1}) = \left(\frac{1 - q^{-1}}{T_s} \right)^2, \quad \psi_2(q^{-1}) = \left(\frac{1 - q^{-1}}{T_s} \right)^4 \quad (31)$$

which can be seen as an acceleration and snap feedforward term. Furthermore, the input shaper is equal to the simulation case study, and the cut-off frequency for the low-pass filter $F(z)$ is set to $f_c = 25$ Hz, which is slightly above the bandwidth of 15 Hz. Next, procedure 5 is executed with initial parameter estimates $\theta(t_0) = [0 \ 0]^T$ and initial matrix $P(t_0) = 10^{-5} \cdot I_{2 \times 2}$. The reference is a fourth-order point-to-point motion as depicted in Fig. 8.

5.3 Experimental results

The experimental results are shown in frequency-domain with a focus on the estimation, and in time-domain to show the performance improvement.

Frequency-domain results: Two experiments are performed, during the first experiment $F = 1$ (—), during the second experiment $f_c = 25$ (—) as proposed earlier. The obtained estimates are compared to the frequency response function (FRF) of the setup, see Fig. 7. This shows that without the use of a low-pass filter a poor estimate is obtained, whereas with the low-pass filter a good estimate of the setup is achieved. It is interesting to note that, although the low-pass filter has a cut-off frequency of $f_c = 25$ Hz the resonance frequency of the plant around 58 Hz is still estimated closely. This is explained using the parametric model in (29) of which the resonance frequency is given by

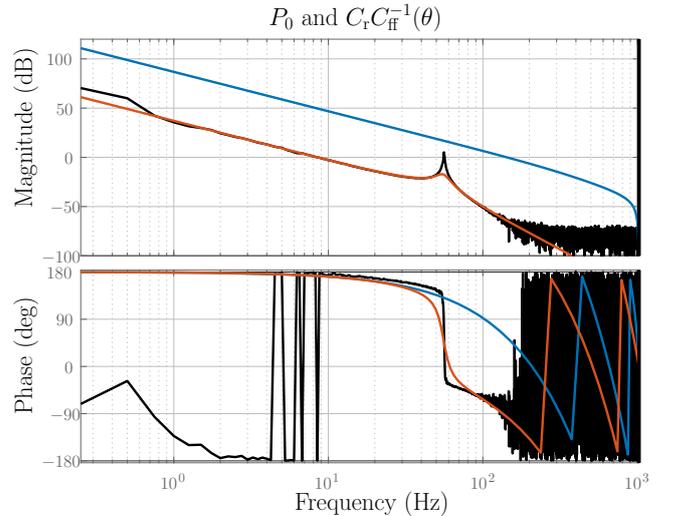


Fig. 7. FRF of the experimental setup (—), and experimental result with $F = 1$ (—) and with $f_c = 25$ Hz (—).

$$\omega_{\text{res}} = \sqrt{\frac{(J_1 + J_2)k}{J_1 J_2}} \quad (32)$$

being a combination of the inertia and stiffness. Hence, by estimating the compliant contributions of the acceleration and snap terms as in (30), the resonance frequency is estimated closely. Furthermore, it is observed that the damping is not properly estimated because it is assumed to be zero and thus not adapted.

Time-domain results: To show the benefit of online learning in the time-domain, also two experiments are performed. During the first experiment only feedback control is used (—) and in the second experiment the proposed feedforward controller optimization is included (—), see Fig. 8. The results show a significant performance improvement. Within a fraction of the first task, about 0.1 seconds, the algorithm has computed the feedforward parameters and the error is reduced with a factor 10, indicating the benefit of direct learning.

To conclude, a good and fast estimate of the feedforward parameters is obtained through current-iteration learning in a practical situation, furthermore the overall performance is improved with a factor 10.

6. CONCLUSIONS & ONGOING RESEARCH

A framework for current-iteration feedforward learning with basis functions is proposed, enabling fast learning while being flexible to task variations. Feedforward parameters are optimized using recursive least squares, and additional filtering is proposed to mitigate the effect of noise and sensor quantization. The method is successfully validated in simulation and an experimental motion control case study is performed. The results show the immediate benefit of learning within a task resulting in a major performance improvement. Furthermore, the results in this paper confirm the theoretical conclusions in Mooren et al. (2019). Ongoing research focuses on fundamental bias elimination along the lines in Blanken et al. (2017).

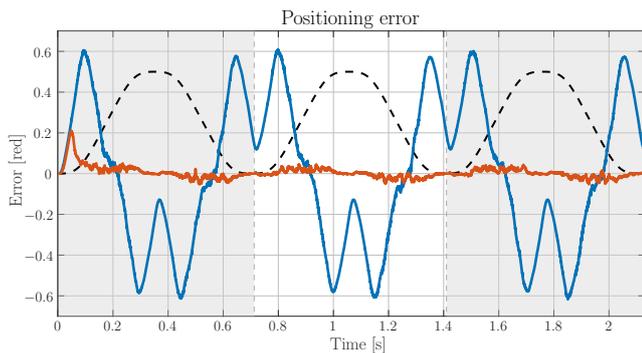


Fig. 8. Positioning error obtained with feedback (—) compared to feedback and learning feedforward (—), and the scaled reference (---).

REFERENCES

Åström, K.J. and Wittenmark, B. (2013). *Adaptive Control: Second Edition*. Dover Publications, INC. Mineola, New York.

Blanken, L., Boeren, F., Bruijnen, D., and Oomen, T. (2017). Batch-to-batch rational feedforward control: From iterative learning to identification approaches,

with application to a wafer stage. *IEEE/ASME Transactions on Mechatronics*, 22(2), 826–837.

Boeren, F., Bareja, A., Kok, T., and Oomen, T. (2016). Frequency-domain ILC approach for repeating and varying tasks: With application to semiconductor bonding equipment. *IEEE/ASME Transactions on Mechatronics*, 21(6), 2716–2727.

Boeren, F., Bruijnen, D., van Dijk, N., and Oomen, T. (2014). Joint input shaping and feedforward for point-to-point motion: Automated tuning for an industrial nanopositioning system. *Mechatronics*.

Boeren, F., Oomen, T., and Steinbuch, M. (2015). Iterative motion feedforward tuning: A data-driven approach based on instrumental variable identification. *Control Engineering Practice*, 37, 11 – 19.

Boerlage, M., Steinbuch, M., Lambrechts, P., and van de Wal, M. (2003). Model-based feedforward for motion systems. In *Proceedings of 2003 IEEE Conference on Control Applications*, volume 2, 1158–1163. IEEE.

Bolder, J., van Zundert, J., Koekebakker, S., and Oomen, T. (2017). Enhancing flatbed printer accuracy and throughput: Optimal rational feedforward controller tuning via iterative learning control. *IEEE Transactions on Industrial Electronics*, 64(5), 4207–4216.

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

Butler, H. (2012). Adaptive feedforward for a wafer stage in a lithographic tool. *IEEE Transactions on Control Systems Technology*.

Gao, X. and Mishra, S. (2014). An iterative learning control algorithm for portability between trajectories. In *2014 American Control Conference*, 3808–3813.

Goodwin, G.C. and Sin, K.S. (2014). *Adaptive filtering prediction and control*. Courier Corporation.

Ho, D. and Enqvist, M. (2018). On the equivalence of forward and inverse IV estimators with application to quadcopter modeling. *IFAC Papers Online*, 51(15), 951 – 956. 18th IFAC Symposium on System Identification.

Hoelzle, D., Alleyne, A., and Johnson, A.W. (2011). Basis task approach to iterative learning control with applications to micro-robotic deposition. *IEEE Transactions on Control Systems Technology*, 19(5), 1138–1148.

Ljung, L. (1999). *System Identification: Theory for the User*. Prentice Hall information and system sciences series. Prentice Hall PTR.

Mooren, N., Witvoet, G., and Oomen, T. (2019). Feedforward motion control: From batch-to-batch learning to online parameter estimation. In *2019 American Control Conference (ACC)*. IEEE (Accepted for publication).

van de Wijdeven, J. and Bosgra, O. (2010). Using basis functions in iterative learning control: analysis and design theory. *Int. Journal of Control*, 83(4), 661–675.

van den Hof, P.M. and Schrama, R.J. (1994). Identification and control-closed loop issues. *IFAC Proceedings Volumes*, 27(8), 311–323.

van der Meulen, S., Tousain, R., and Bosgra, O. (2008). Fixed structure feedforward controller design exploiting iterative trials: Application to a wafer stage and a desktop printer. *Journal of Dynamics Systems, Measurements and Control*, 130.

Widrow, B. (1987). Adaptive inverse control. In *Adaptive Systems in Control and Signal Processing 1986*, 1–5. Elsevier.