Data-Driven Multivariable ILC: Enhanced Performance by Eliminating $L$ and $Q$ Filters

Joost Bolder, Stephan Kleinendorst, and Tom Oomen

Eindhoven University of Technology, Dept. of Mechanical Engineering, Control Systems Technology group, Den Dolech 2, Eindhoven, The Netherlands

SUMMARY

Iterative learning control algorithms enable high performance control design using only approximate models of the system. To deal with severe modeling errors, a robustness filter $Q$ is typically employed. Irrespective of the large performance enhancement, these approaches thus require a modeling effort and are subject to a performance/robustness tradeoff. The aim of this paper is to develop a fully data-driven ILC approach that does not require a modeling effort and mitigates the performance/robustness tradeoff. The main idea is to replace the use of a model by dedicated experiments on the system. Convergence conditions are developed in a finite-time framework and insight in the convergence aspects are presented using a frequency domain analysis. Extensions to increase the convergence speed are proposed. The developed framework is validated through experiments on a multivariable industrial flatbed printer. Both increased performance and robustness are demonstrated in a comparison with closely related model-based ILC algorithms. Copyright © 0000 John Wiley & Sons, Ltd.

KEY WORDS: Iterative Learning Control; Optimal ILC; Data-driven control

1. INTRODUCTION

Iterative learning control (ILC) [1] can significantly enhance the performance of systems that perform repeated tasks. After each repetition, the control signals for the next repetition are updated by learning from past experiments, see the block diagram in Fig. 1. Examples applications include additive manufacturing machines [2, 3], robotic arms [4], printing systems [5], pick-and-place machines, electron microscopes, wafer stages [6–8], and nuclear fusion reactors [9].

Iterative learning control algorithms including frequency domain ILC [10, 11], optimal ILC [12–17], and Arimoto-type algorithms [18] are to a certain extent model-based. Indeed, the convergence conditions and performance properties of these learning control algorithms hinge on model knowledge of the controlled system. In view of achieving perfect performance, the learning filter, typically denoted as $L$, is usually based on an approximate model where model errors up to 100% are tolerable, see e.g., [11, 19]. Due to the inherent approximate nature of models, the model of any physical system typically has a large model error. For example, in mechanical systems, if a resonance is missed, then both magnitude and phase have a large error with respect to the true system, see [20, Section 7.4.6]. This necessitates the design of a robustness filter, often denoted as $Q$, either manually or through robust ILC approaches including [21–24].

Although there are substantial developments in robust ILC, such approaches drastically increase the modeling requirements and lead to a performance/robustness tradeoff. In particular, these...
approaches require both a nominal model and a description of model uncertainty. Especially in the multivariable situation, such models are difficult and expensive to obtain. In addition, when the uncertainty is too large, the required robustness leads to a poor performance. The aim of this paper is to develop an optimal ILC algorithm for multivariable systems in which the need for a model and the performance/robustness tradeoff are eliminated. In fact, the proposed approach will have no learning and robustness filters in the usual ILC sense.

Other data-driven ILC approaches have been developed, aiming to address the fact that a model is required in the design of the \( L \) and \( Q \) filters. In \([25–28]\), a model is estimated after each trial and used for ILC. As such, these approaches essentially classify to a traditional model-based ILC approach, involving a recursive or adaptive estimation scheme. The methods are hence also subject to a performance/robustness tradeoff, depending on how well the estimated model captures the true system.

In the present paper, the need for a learning filter is replaced by dedicated experiments on the true system. In these experiments, the gradient of a performance criterion is directly measured and used for the learning update. The main difficulty in the development of the presented approach lies in the multivariable aspect. Indeed, when the proposed approach is applied to the special case of SISO systems, a well-known result is recovered that is closely related to the commonly used “FiltFilt” \([29]\) approach in robustness filtering \([30, \text{Section 36.3.3.1}]\). This standard solution for SISO systems is also well-known and commonly applied in system identification \([31, 32, 33, \text{Section 12.2}]\), in ILC \([34–38]\) and recently also exploited in virtual reference feedback tuning \([39]\).

The main contribution of this paper lies in a fully data-driven optimal ILC framework for multivariable systems. This is achieved by an approach that resembles recent results in system identification \([40]\). The main contributions are

1. the development of a data-driven ILC algorithm with convergence conditions,
2. insight in the convergence aspects is obtained using a frequency-domain analysis,
3. an extension to enhance the convergence speed that relies on a data-driven quasi-Newton approach,
4. connections with closely related model-based ILC algorithms are established,
5. experimental validation on an industrial multivariable flatbed printer.

Preliminary research related to contribution 1 appeared in \([41]\). The present paper extends \([41]\) with contributions 2 – 5 and more theory and explanations.

The outline of this paper is as follows. In the next section, the problem is stated and the contributions are summarized. Then, in Section 2, the data-driven ILC algorithm and convergence conditions are developed. In Section 3, a frequency-domain interpretation of the convergence aspects is presented, leading to the extensions to increase convergence speed in Section 4. Several connections with common pre-existing ILC algorithms are established in Section 5. The results of the present paper are supported with an experimental validation on industrial multi-axis flatbed printer in Section 6. The conclusions and ongoing research topics are presented in Section 7. Finally, implementation aspects are elaborated on in the appendices.
Notation Systems are considered in discrete time. The spectral radius of a matrix $A \in \mathbb{R}^{N \times N}$ is given by $\lambda(A) = \max_{1 \leq i \leq N} |\lambda_i(A)|$, with $\lambda(A) = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ the spectrum of $A$. A matrix $B \in \mathbb{R}^{N \times N}$ is defined positive definite if $x^T B x > 0, \forall x \neq 0 \in \mathbb{R}^N$ and is denoted as $B \succ 0$. A matrix $C \in \mathbb{R}^{N \times N}$ is defined positive semi-definite if $x^T C x \geq 0, \forall x \in \mathbb{R}^N$ and is denoted as $C \succeq 0$.

For a vector $x$, the weighted 2-norm is denoted as $||x||_W := \sqrt{x^T W x}$, where $W$ is a weighting matrix.

In this section, the proposed approach is developed. This constitutes contribution 1, see Section 1.

2.1. System description

Consider a single-input single-output (SISO) causal system $J_{11}$ with the corresponding transfer function denoted as

$$J_{11}(z) = \sum_{i=0}^{\infty} h_i z^{-i},$$

where $h_i \in \mathbb{R}, i = 0, \ldots, \infty$ are the Markov parameters of $J_{11}$, and $z$ is a complex indeterminate which is sometimes tacitly omitted for conciseness. It is assumed that signals have finite length $N \in \mathbb{N}$. The response of the system $y^1 = J_{11} f^1$ for the finite-time interval $0 \leq k < N$ is denoted as

$$\begin{bmatrix} y^1(0) \\ y^1(1) \\ \vdots \\ y^1(N-1) \end{bmatrix} = \begin{bmatrix} h(0) & 0 & \ldots & 0 \\ h(1) & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ h(N-1) & \ldots & h(1) & h(0) \end{bmatrix} \begin{bmatrix} f^1(0) \\ f^1(1) \\ \vdots \\ f^1(N-1) \end{bmatrix},$$

with $y^1 \in \mathbb{R}^N, f^1 \in \mathbb{R}^N$, and $J_{11} \in \mathbb{R}^{N \times N}$ a finite-time matrix representation of $J_{11}(z)$.

Consider a multiple-input multiple-output (MIMO) system $J$ with transfer function matrix $J(z) \in \mathbb{C}^{n_o \times n_i}$, with $n_i$ the number of inputs, $n_o$ the number of outputs. The finite-time response for the MIMO system $J$ is denoted as

$$\begin{bmatrix} y^i \\ \vdots \\ y^{n_o} \end{bmatrix} = \begin{bmatrix} J_{i1} & \ldots & J_{i,n_i} \\ \vdots & \ddots & \vdots \\ J_{n_o,1} & \ldots & J_{n_o,n_i} \end{bmatrix} \begin{bmatrix} f^1 \\ \vdots \\ f^{n_i} \end{bmatrix},$$

where $J_{ij}$ is the matrix representation of the $ij^{th}$ entry in $J(z), y^i \in \mathbb{R}^N, f^i \in \mathbb{R}^N, y \in \mathbb{R}^{n_o,N}, f \in \mathbb{R}^{n_i,N}$, and $J \in \mathbb{R}^{n_o \times n_i \times N}$ is the matrix representation of $J(z)$.

2.2. Optimal adjoint-based ILC

The ILC framework used in this paper is presented in Fig. 2. The system $J \in \mathbb{R}^{n_o,N \times n_i,N}$ is a MIMO system with output $y_j \in \mathbb{R}^{n_o,N}$, input $f_j \in \mathbb{R}^{n_i,N}$ and reference $r \in \mathbb{R}^{n_o,N}$. The trial index is denoted...
The gradient descent ILC algorithm that minimizes (3) is given by

\[ f_{j+1} = (I - \varepsilon W_f) f_j + \varepsilon J^T W_e e_j, \quad 0 \leq \varepsilon \leq 0 \]

where the factor 2 is absorbed in the step size \( \varepsilon \). Note that the gradient descent direction (5) does not depend on \( W_{\Delta f} \). Rearranging (6) leads to a gradient-descent ILC algorithm for minimizing (3) and is presented next.

**Algorithm 2 (Gradient descent ILC)**

The gradient descent ILC algorithm that minimizes (3) is given by

\[ f_{j+1} = (I - \varepsilon W_f) f_j + \varepsilon J^T W_e e_j, \quad 0 \leq \varepsilon \leq 0 \]

An upper bound on the learning gain \( \varepsilon \leq \varepsilon \) to ensure convergence of algorithm (7) is developed later, see Theorem 7 in Section 2.4. In the following, it is shown that \( J^T \) has the interpretation of the adjoint operator of \( J \), see also [42, Section 22].

Figure 2. Multivariable ILC setup with MIMO system \( J \), output \( y_j \), input \( f_j \), reference \( r \), and tracking error \( e_j \). The trial index is denoted as \( j \).
Definition 3 (Adjoint)
Let the inner product of two signals be given by: $\langle u, g \rangle = u^T g$, with $u, g \in \mathbb{R}^N$. Then, for a linear operator $J$, the adjoint $J^*$ is defined as the operator that satisfies the condition

$$\langle f, Jg \rangle = \langle J^* f, g \rangle \forall f, g \in \mathbb{R}^N.$$ 

The adjoint $J^*$ of $J$, see (1), is given by $J^* = J^T$. Indeed, from Definition 3 follows that if $J^*$ is the adjoint of $J$ then

$$f^T Jg = (J^* f)^T g = f^T (J^*)^T g, \forall f, g \in \mathbb{R}^N.$$ 

This reveals that $J^T$ is the adjoint of $J$. Therefore, gradient descent ILC Algorithm 2 has the interpretation of an adjoint-based ILC algorithm.

2.3. Data-driven learning using the adjoint system

In this section, a data-driven approach is presented that enables filtering through an adjoint of a multivariable system. In the well-known SISO case, an operation with the adjoint of a linear time invariant SISO system can be recast to an operation on the original system and time-reversal of the in- and output signals. There are many applications that exploit this property, e.g., system identification [31, 32, 40], [33, Section 12.2] and iterative learning control [30, Section 36.3.3.1], [28, 34–38, 43]. The generalization to MIMO systems requires significantly more steps and is investigated next. The theory developed here resembles recent developments in system identification [40].

Before presenting the main results for MIMO systems, note that for a SISO system $J^{11}$, the adjoint $J^{11T}$ can be recast to

$$J^{11T} = \mathcal{T} J^{11} \mathcal{T}$$

with

$$\mathcal{T} = \begin{bmatrix} 0 & \ldots & 0 & 0 & 1 \\ 0 & \ldots & 0 & 1 & 0 \\ \vdots & \ddots & 0 & 0 \\ 0 & \ddots & \vdots & \vdots \\ 1 & 0 & \ldots & 0 & 0 \end{bmatrix}$$

an involutory permutation matrix with size $N \times N$. Note that this step is valid since $J^{11}$ has a lower-triangular Toeplitz structure, since it is a linear time-invariant system. Here, $\mathcal{T}$ has the interpretation of a time-reversal matrix. Next,

$$y = J^{11T} f = \mathcal{T} J^{11} \mathcal{T} f,$$

which shows that the operation $y = J^{11T} f$ can be recast to an operation on the original system $J^{11}$, and time-reversing the input before applying operator $J^{11}$, and time-reversing the resulting output afterwards. Note that (8) is only valid for SISO systems and is in fact exactly the same operation as is performed using the “FiltFilt” [29] operation in robustness filtering [30, Section 36.3.3.1].

In case of a MIMO system $J$, the adjoint $J^T$ can be written as

$$J^T = \begin{bmatrix} \mathcal{T} & 0 & \ldots & 0 \\ \mathcal{T} & J^{11} & \ldots & J^{n_11} \\ \vdots & \ldots & \vdots & \vdots \\ \mathcal{T} & J^{n_{n_1}1} & \ldots & J^{1n} \end{bmatrix} \begin{bmatrix} \mathcal{T} & 0 & \ldots & 0 \\ \mathcal{T} & \mathcal{T} & J^{11} & \ldots & J^{n_11} \\ \vdots & \ldots & \vdots & \vdots & \vdots \\ \mathcal{T} & \mathcal{T} & \ldots & \ldots & \ldots \end{bmatrix}$$

(9)
where $\tilde{J} \in \mathbb{R}^{n_i \times n_o \times N}$, and $\mathcal{T}_{n_i} \in \mathbb{R}^{n_i \times n_i \times N}$, $\mathcal{T}_{n_o} \in \mathbb{R}^{n_o \times n_o \times N}$ are time-reversal operators for the higher dimensional in- and output signals. Matrix $\tilde{J}$ is the finite-time representation of $\tilde{J}(z) \in \mathbb{C}^{n_i \times n_o}$, with

$$
\tilde{J}(z) = \begin{bmatrix}
J^{11}(z) & \cdots & J^{1n_o}(z) \\
\vdots & \ddots & \vdots \\
J^{n_i1}(z) & \cdots & J^{n_in_o}(z)
\end{bmatrix}.
$$

The key observation is that $\tilde{J} \neq J$ for general MIMO systems. Indeed, the standard approach as in the SISO case can only be applied if $\tilde{J} = J$, i.e., $J$ must be a symmetric system.

The main idea of the presented approach is to develop a data-driven MIMO ILC algorithm by noting that

$$
\tilde{J}(z) = \sum_{i=1}^{n_i} \sum_{j=1}^{n_o} E_{ij} J(z) E_{ij},
$$

where $E_{ij} \in \mathbb{R}^{n_i \times n_o}$, is a static system with $n_i$ outputs and $n_o$ inputs. For the $k^{th}$ and $l^{th}$ entry of $E_{ij}$ holds

$$
E_{ij} = \begin{cases} 1 & \text{if } k = i, l = j \\
0 & \text{otherwise}
\end{cases},
$$

i.e., all entries are of $E_{ij}$ zero, except the $ij^{th}$ entry. The structure of $E_{ij}$ is given by

$$
E_{ij} = \begin{bmatrix}
0^{i-1 \times j-1} & 0^{i-1 \times 1} & 0^{i-1 \times n_o-j} \\
0^{1 \times j-1} & 1 & 0^{1 \times n_o-j} \\
0^{n_i-i \times j-1} & 0^{n_i-i \times 1} & 0^{n_i-i \times n_o-j}
\end{bmatrix},
$$

where $0^{i \times j}$ is the zero matrix with $\dim(0^{i \times j}) = (i, j)$. The role of $E_{ij}$ in $y(z) = J(z) E_{ij} u(z)$ is selecting the $j^{th}$ entry in $u(z)$ and apply it to the $i^{th}$ input of $J(z)$, where the rest of the inputs to $J(z)$ are zero.

Let $E_{ij}$ be the finite-time representation of $E_{ij}$, then the finite-time representation of $\tilde{J}$, see (10), is given by

$$
\tilde{J} = \sum_{i=1}^{n_i} \sum_{j=1}^{n_o} E_{ij} J E_{ij}.
$$

Substitution of the above in (9) yields

$$
J^T = \mathcal{T}_{n_i} \left( \sum_{i=1}^{n_i} \sum_{j=1}^{n_o} E_{ij} J E_{ij} \right) \mathcal{T}_{n_o}.
$$

The above equation recasts the evaluation of $J^T$ as $n_i \times n_o$ experiments on $J$. This approach is used with ILC algorithm (7), to arrive at the data-driven ILC algorithm for MIMO systems.

The following procedure provides the learning update for the ILC Algorithm 2.

**Procedure 4 (Dedicated gradient experiment)**

The objective is to compute $J^T e_j$ by performing experiments on $J$. This is achieved by applying the following sequence of steps.

1. Time reverse $\mathcal{T}_{n_o} e_j$. 
2. Compute $\overline{e}_j$ by performing $n_i \cdot n_o$ experiments on $J$ using

$$\overline{e}_j = \sum_{i=1}^{n_i} \sum_{j=1}^{n_o} E^{ij} J E^{ij} \overline{e}_j,$$

with $E^{ij}$ the matrix representation of static system $E^{ij}$ defined in (11).

3. Time reverse again to compute $J^T e_j = T_{n_i} \overline{e}_j$.

In traditional ILC $J$ is a model, in which case Procedure 4 can directly be used. However, the main idea here is that Procedure 4 only depends on $J$ and not $J^T$, hence the learning update in (7) can be accomplished by performing dedicated experiments on the real system instead of using models. The above approach is visualized in a diagram for $n_i = n_o = 2$ in Fig. 3.

The complete adjoint-based ILC algorithm (7) using the data-based learning update in (12) is presented next.

**Procedure 5** (MIMO data-driven adjoint-based ILC)

Given an initial input $f_0$, set $j = 0$, perform the following steps.

1. Perform an experiment and measure $e_j = r - J f_j$.

2. Use Procedure 4 to experimentally determine $v_j = J^T W e_j$.

3. Apply ILC algorithm (7), set $f_{j+1} = (I - \varepsilon W_e) f_j + \varepsilon v_j$.

4. Set $j := j + 1$ and go back to step 1 or stop if a suitable stopping criterion is met.

Note that the inclusion of weighting matrix $W_e$ in step 2 of Procedure 5 is a straightforward multiplication of $e_j$ prior to using Procedure 4. A suitable stopping criterion can be a threshold on the cost function value resulting from (3), or a threshold on the change in cost function value.

**Remark 1**

In case MIMO system $J$ is (almost) symmetric, i.e., $J \approx J^T$, only a single experiment is needed to experimentally determine $J^T W_e e_j$ in step 2 in Procedure 5 since $J^T W_e e_j \approx J W_e e_j$. Standard robustness filtering, see e.g. [1], could be applied to accommodate for differences between $J$ and $J^T$.

**Remark 2**

All measured signals contain noise. In principle, to show that Procedure 4 yields unbiased results requires computing the expected value of the filtered $e_j$. A full stochastic proof that $J^T e_j$ resulting from Procedure 4 is indeed unbiased follows along the lines of [44] and is outside the scope of this paper.

Copyright © 0000 John Wiley & Sons, Ltd. *Int. J. Robust. Nonlinear Control* (0000) DOI: 10.1002/rnc
2.4. Convergence conditions

In this section, the convergence of Procedure 5 is analyzed. Firstly, monotonic convergence is defined.

**Definition 6** (Monotonic convergence)

The ILC algorithm is monotonically convergent in the 2-norm if

\[ \|f_{j+1} - f_\infty\| \leq \gamma \|f_j - f_\infty\|, \quad \forall f_j, f_\infty \]  \hspace{1cm} (13)

with \( f_\infty \) the unique fixed point of iterative algorithm (7) and \( \gamma \in [0, 1) \) a convergence rate.

See [1] and [11] for equivalent definitions. Next, the conditions for achieving monotonic convergence of ILC algorithm (7) are presented.

**Theorem 7** (Condition for monotonic convergence)

Given weighting matrices \( W_e, W_f \), such that \( J^TW_eJ + W_f \succ 0 \), then for any \( \varepsilon \leq \bar{\varepsilon} \) with

\[ \bar{\varepsilon} = 2\|J^TW_eJ + W_f\|^{-1}, \]  \hspace{1cm} (14)

algorithm (7) is monotonically convergent. In addition,

\[ f_\infty = \lim_{j \to \infty} f_j = (J^TW_eJ + W_f)^{-1}J^TW_e r, \]  \hspace{1cm} (15)

\[ e_\infty = \lim_{j \to \infty} e_j = (I - J(J^TW_eJ + W_f)^{-1}J^TW_e) r. \]  \hspace{1cm} (16)

**Proof** Substituting (2) in (7) yields

\[ f_{j+1} = (I - \varepsilon(J^TW_eJ + W_f))f_j + \varepsilon J^TW_e r, \]  \hspace{1cm} (17)

with \( M = I - \varepsilon(J^TW_eJ + W_f) \) and \( m = \varepsilon J^TW_e r \). Note that \( \varepsilon(J^TW_eJ + W_f) \) is a real-symmetric matrix and has real eigenvalues. Hence (17) is a stable system with \( \lambda(M) < 1 \) iff \( \lambda(\varepsilon(J^TW_eJ + W_f)) < 2 \). Again, since \( \varepsilon(J^TW_eJ + W_f) \) is a real symmetric matrix, the spectral radius and maximal singular value coincide, hence results \( \|\varepsilon(J^TW_eJ + W_f)\| < 2 \iff \lambda(M) < 1 \iff \|M\| < 1 \), yielding \( \varepsilon \leq \bar{\varepsilon} \) with \( \bar{\varepsilon} = 2\|J^TW_eJ + W_f\|^{-1} \).

From (17) results \( f_\infty = (I - M)^{-1} m \) using \( f_{j+1} = f_j = f_\infty \), proving (15). Similar steps yield \( e_\infty \) in (16). Next, it can be shown that

\[ \|f_{j+1} - f_\infty\| = \|M_{f_j} - M_{f_\infty}\| \]

by substituting (17) in the left hand side of (13) and substituting \( f_\infty = M_{f_\infty} + m \). Using \( \|M_{f_j} - M_{f_\infty}\| \leq \|M\|\|f_j - f_\infty\| \), it follows that if and only if \( \|M\| < 1 \), then condition (13) for monotonic convergence is satisfied \( \forall f_j, f_\infty \) with \( \gamma = \|M\| \).

For further results in related adjoint-based ILC algorithms, see, e.g., [36] and [17]. From Theorem 7 results an upper-bound for the maximal learning gain that ensures convergence of the ILC algorithm. This upper-bound depends on the system \( J \), see (14).

**Remark 3** (Robustness against model uncertainties)

The developed approach does not require the design of a learning filters in contrast to classical approaches as in [1]. In classical ILC design, all modeling errors are typically addressed through a robustness \( Q \)-filter design, see [30], which goes at the expense of performance. In this sense, the presented approach eliminates performance/robustness tradeoff in ILC designs.

In the next section, the relation between \( J \) and the convergence speed is investigated using a frequency-domain analysis.
3. FREQUENCY-DOMAIN ANALYSIS

In this section, the adjoint-based ILC algorithm in Procedure 5 is analyzed in the frequency domain, constituting contribution 2, see Section 1. The results show that the convergence speed for adjoint-based ILC algorithms can be directly related to the frequency response characteristics of $J(e^{j\omega})$. The results are illustrated with an example where it is shown that slow convergence of adjoint-based ILC algorithms is to be expected when $J(e^{j\omega})$ has large ratio between the maximal and minimal magnitude.

To facilitate a frequency-domain analysis, assume $W_f = 0$ and $W_o = I$ in (2). Next, given adjoint system $J^*(e^{j\omega})$, then a frequency-domain adjoint-based ILC algorithm is given by

$$f_{j+1}(e^{j\omega}) = f_j(e^{j\omega}) + \varepsilon J^* e_j(e^{j\omega}), \quad (18)$$

with $J^*(e^{j\omega}) = J^T(e^{-j\omega})$ for $J(e^{j\omega}) \in \mathbb{C}^{n_o \times n_i}$. The trial dynamics follow by substituting $e_j = r - J f_j$ in (18) and rearranging

$$f_{j+1} = (I - \varepsilon J^* J) f_j + \varepsilon J^* r.$$

A frequency domain convergence condition follows along identical lines as in Theorem 7, see also [11], and is given in the following corollary.

Corollary 8 (Frequency-domain convergence criterion)
Algorithm (18) is monotonically convergent in the sense of Definition 6, replacing $f_{j+1}$ with $f_{j+1}$, $f_j$ with $f_j$, and $\gamma$ with $\gamma$ in (13), if

$$||M(e^{j\omega})|| < 1, \forall \omega,$$

with $M(e^{j\omega}) = 1 - \varepsilon J^*(e^{j\omega})J(e^{j\omega})$.

Using this convergence condition yields monotonic convergence with $\gamma = ||M(e^{j\omega})||$, see Definition 6 and the proof of Theorem 7. Next, a Bode magnitude condition is developed which is commonly used in ILC for SISO systems. Consider a SISO system $J^{11}(e^{j\omega})$. For SISO systems, the adjoint reduces to $J^{11*}(e^{j\omega}) = J^{11}(e^{-j\omega})$. Consequently, the convergence condition in Corollary 8 reduces to

$$|M^{11}(e^{j\omega})| < 1, \forall \omega, \quad (19)$$

with $M^{11}(e^{j\omega}) = 1 - \varepsilon J^{11*}(e^{-j\omega})J^{11}(e^{j\omega})$. Note that $J^{11}(e^{-j\omega})J^{11}(e^{j\omega})$ is a real-valued function and that $|J^{11*}J^{11}| = |J^{11}|^2$. Hence, the convergence condition in (19) is satisfied if $\varepsilon < \varepsilon$ with

$$\varepsilon = \sup_{\omega} \frac{2}{|J^{11}(e^{j\omega})|^2}. \quad (20)$$

It shows that the maximal allowable $\varepsilon$ is determined by the the peak value of the frequency response function of $J^{11}$.

Next, note that $|M^{11}(e^{j\omega})|$ in (19) has an interpretation in terms of a frequency-dependent update-rate of $f_{j+1}(e^{j\omega})$. For a particular frequency $\omega^*$, a value $|M^{11}(e^{j\omega^*})| \approx 0$ results in a large trial-to-trial change in $f_{j+1}(e^{j\omega^*})$, and a value $|M^{11}(e^{j\omega^*})| \approx 1$ results in a very small change. The shape of $|M^{11}(e^{j\omega})|$ is determined by the system $J^{11}(e^{j\omega})$ and by the learning gain $\varepsilon$. Since the maximal allowable learning gain $\varepsilon$ directly results from the peak value of $J^{11}(e^{j\omega})$, the shape of the frequency response function essentially determines the overall convergence behavior. In case the ratio between the peak value and the minimal value of $|J^{11}(e^{j\omega})|$ is large, slow convergence should be expected. This is illustrated with the following example.

Example 9
Consider the second-order system

$$J^{11}(z) = \frac{4.29z^2 - 8.53z + 4.25}{z^2 - 1.98z + 0.99}, \quad (21)$$
with the sampling frequency $f_s = 1$ Hz. A Bode diagram of $J^{11}$, $J^{11^*}$, and $J^{11^*} J^{11}$ is presented in Fig. 4. As is expected, analysis of Fig. 4 reveals that $J^{11^*}$ has the inverse phase of $J^{11}$, hence in the product $J^{11^*} J^{11}$, the phases cancel and the resulting phase of $J^{11^*} J^{11}$ is zero.

The peak value of $|J^{11}(e^{i\omega})|$ is shown in Fig. 4 with the dashed gray line, i.e., $\sup_{\omega} J^{11^*} J^{11} = 60.7$ dB. From the peak value results a maximal learning gain $\bar{\epsilon} = 1.8 \cdot 10^{-4}$ using (20). Let $\epsilon = 0.5 \bar{\epsilon} = 9.0 \cdot 10^{-5}$ such that (19) is satisfied for example system (21).

The resulting update rate $|M^{11}(e^{i\omega})|$ is also shown in Fig. 4. The results show that for frequencies around the peak value of $|J^{11}(e^{i\omega})|$, large trial-to-trial changes in $f_j(e^{i\omega})$ are obtained, see the very small values of $|M^{11}(e^{i\omega})|$ at $f \approx 1.6 \cdot 10^{-2}$ Hz.

On the other hand, $|M^{11}(e^{i\omega})|$ is close to 1 for almost all other frequencies. Indeed, the system includes a resonance which determines the peak-value. Clearly, $|J^{11}(e^{i\omega})|$ peaks in only a small frequency neighborhood around the resonance, and the rest of $|J^{11}(e^{i\omega})|$ is significantly lower. This illustrates that the gradient-descent ILC algorithm converges slowly for this example system since the ratio between peak and minimal value of $J^{11}(e^{i\omega})$ is large.

Similar phenomena as in the example system in Fig. 4 are often present in mechanical systems. Using gradient-descent Algorithm 2 may yield slow convergence in this case. Also in [45], slow convergence of adjoint-based ILC algorithms is reported for non-minimum phase systems.

In the next section, extensions to (7) are presented to increase the convergence speed significantly.

4. ENHANCING CONVERGENCE SPEED

In the previous section, a frequency domain analysis is employed to gain insight in the convergence speed of adjoint-based ILC algorithms. In this section, algorithm (7) is extended using concepts from optimization theory in order to enhance the convergence speed. This constitutes contribution 3 and 4 in Section 1. Several common ILC algorithms including standard norm-optimal ILC, see e.g., [12,14] and the algorithm in [17] termed parameter-optimal ILC are recovered as special cases. Moreover, the results presented here are suitable for fully data-driven implementation on MIMO systems.
4.1. Incorporating Hessian information

The main idea of gradient descent ILC algorithm (7) is taking small steps towards the minimum of (3). As illustrated in Section 3, the maximal step size is very small if the dynamic range of the system’s response is large. The use of Hessian information, i.e., knowledge of the Hessian of (3), can significantly improve convergence speed. In fact, by setting \( \frac{\partial J(f_{j+1})}{\partial f_{j+1}} = 0 \) in (4), the optimal solution to (3) can be obtained in a single step and is given by

\[
\arg\min \ J(f_{j+1}) = f_j - \left( \frac{\partial^2 J(f_{j+1})}{\partial f_{j+1}^2} \right)^{-1} \left( \frac{\partial J(f_{j+1})}{\partial f_{j+1}} \right)_{f_{j+1}=f_j} .
\]

This general solution is well known and used in Newton’s method for optimization, see e.g., [46, Section 8.8]. The following ILC algorithm extends the algorithm in (7) and encompasses (22).

**Algorithm 10**  (Data-driven quasi-Newton ILC)

Given learning gain \( \varepsilon_j \in \mathbb{R} \) and matrix \( B_j \in \mathbb{R}^{n_o \times n_i} \), the quasi-Newton ILC algorithm is given by

\[
f_{j+1} = (I - \varepsilon_j B_j W_f) f_j + \varepsilon_j B_j J^T W_e e_j , \quad (23)
\]

Clearly, gradient-descent ILC algorithm in (6) is recovered by setting \( \varepsilon_j = \varepsilon \) and \( B_j = I \) in (23).

**Algorithm 10** is inspired by (22) where the inverse Hessian \( \frac{\partial^2 J(f_{j+1})}{\partial f_{j+1}^2}^{-1} \) is replaced with an estimate \( B_j \). A procedure to determine an estimate \( B_j \) from already available measurement data is presented later. First, consider the following theorem, where convergence in a single iteration using **Algorithm 10** is illustrated in case estimate \( B_j \) is exact.

**Theorem 11**

Set \( W_{\Delta f} = 0 \) in (3), let \( \varepsilon_j = 1 \) and set \( B_j = (J^T W_e J + W_f)^{-1} \), then ILC algorithm (23) directly minimizes performance criterion (3) and also achieves \( f_{j+1} = f_{\infty} \) in a single iteration.

**Proof** From Theorem 7 follows that \( f_{\infty} = (J^T W_e J + W_f)^{-1} J^T W_e r \), which is also the minimizer of (3) since \( W_{\Delta f} = 0 \). Substituting (2) in (23) yields \( f_{j+1} = f_{\infty} \forall f_j \).

In **Theorem 11**, \( B_j \) is the inverse Hessian of (3). If Hessian knowledge is available, much faster convergence then gradient descent algorithm (7) may be achieved. **Theorem 11** also shows that the Hessian is independent of \( f_j \), since (3) is a quadratic criterion. In fact, Theorem 7 recovers standard norm-optimal ILC as developed in, e.g., [14]. To arrive at a fully data-driven implementation of Algorithm 10, consider the following procedure to estimate \( B_j \) and compute a suitable \( \varepsilon_j \).

**Procedure 12**  (Data-driven Hessian estimation)

Given \( B_{j-1}, f_{j-1}, f_j, J^T W_e e_{j-1} \) and \( J^T W_e e_j \) resulting from step 2 in Procedure 5, then \( B_j \) in (23) is given by

\[
B_j = B_{j-1} - \frac{\Delta_j \zeta_j^T B_j + B_j \zeta_j \Delta_j^T}{\Delta_j \zeta_j} + \left( 1 + \frac{\zeta_j^T B_j \zeta_j}{\Delta_j \zeta_j} \right) \frac{\Delta_j \Delta_j^T}{\Delta_j \zeta_j} , \quad (24)
\]

where

\[
\Delta_j = f_j - f_{j-1}, \quad \zeta_j = w_j - w_{j-1},
\]

with \( w_j = 2 J^T W_e e_j - 2 W_f f_j \). The learning gain \( \varepsilon_j \) follows by minimizing (3) over \( \varepsilon_j \) given \( f_{j+1} \) in (23). Note that (23) is affine in \( \varepsilon_j \) which enables the following analytic solution

\[
\varepsilon_j = \arg\min \ J( f_{j+1}(\varepsilon_j) ) = \frac{||w_j||_{B_j}^2}{||J B_j w_j||_{W_e}^2 + ||B_j w_j||_{W_f}^2 + ||B_j w_j||_{W_{\Delta f}}^2} \quad (25)
\]

with \( f_{j+1}(\varepsilon_j) \) given in (23) and \( B_j \) resulting from (24).
Procedure 12 is known as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) approach for constructing the inverse Hessian, see e.g., [46, Section 10.4] combined with a commonly used line-search, see e.g., [46, Section 8.5]. The BFGS method combined with quasi-Newton-type algorithms is known to be effective in practice, and is generally considered one of the best general purpose methods for unconstrained optimization problems, see [47] for the convergence aspects. The main idea is that the gradient information that is already available is used to generate curvature information of the cost function. The key motivation for using this approach is the observation that the Hessian of (3) is invariant under \( f_j \), as shown in Theorem 11. It is hence expected that this approach also works well for the data-driven ILC approach in the present paper.

Note that \( B_j \) can be directly computed using the measurement data that results from step 3 in Procedure 4. Using an optimal step size \( \varepsilon_j \) as in (25) is often used to improve the convergence properties. Indeed, the accuracy of \( B_j \) resulting from (24) is unknown a priori. Using a fixed \( \varepsilon \) as in Theorem 7 may achieve convergence since the BFGS estimate inhibits strong self-correcting properties, see [47]. On the other hand, determining \( \varepsilon_j \) from (25) generally results in a decrease of \( J \) for all trials, which is not guaranteed with a fixed \( \varepsilon \). It is hence expected that using a trial-varying \( \varepsilon_j \) both enhances the convergence speed and robustness of the extended data-driven ILC algorithm in (23). Using (25) comes with the expense of one additional experiment in order to determine \( \varepsilon_j \) in (25) since \( JB_jw_j \) is not available form measurement data of Procedure 4. In this experiment the input to the system is set to \( B_jw_j \) from which \( JB_jw_j \) can be directly measured.

The fully data-driven quasi-Newton ILC Algorithm 10 using the Hessian estimation in Procedure 12 is summarized as follows.

**Summary 13**

Given an initial input \( f_0 \) and initial \( B_0 \), set \( j = 0 \), and perform the following steps.

1. Perform a trial and measure \( e_j = r - Jf_j \).
2. Use Procedure 4 to experimentally determine \( v_j = JTWe_j \).
3. Use Procedure 12 to compute \( B_j \) and experimentally determine \( \varepsilon_j \).
4. Apply ILC algorithm (23), set \( f_{j+1} = (I - \varepsilon_jB_jW_f)f_j + \varepsilon_jB_jv_j \).
5. Set \( j := j + 1 \) and go back to step 1 or stop if a suitable stopping condition is met.

**Remark 4**

The inverse Hessian \( B_j \) may also be determined by performing experiments on \( J \) instead of estimating it from data already available, for instance, along the lines of [44]. This is especially relevant in case a non-convex performance criterion is used, and local Hessian information is needed. An example of a non-convex performance criterion in ILC is presented in [48], where extrapolation capabilities with respect to changes in \( r \) are introduced using rational basis functions. Note that since all measured signals contain measurement noise, a direct estimate of the Hessian can be biased, see [49] for the analysis and unbiased estimates.

In the next section, several important ILC approaches are recovered as special cases of the proposed approach.

5. CONNECTIONS WITH COMMON PRE-EXISTING ILC ALGORITHMS

In this section, it is shown that standard norm-optimal ILC, see e.g., [12, 14] and the parameter-optimal ILC algorithm in [16, 17, 50] are special cases of Algorithm (10) and Procedure 12. Norm-optimal ILC is an important class of ILC algorithms, see e.g., [2, 12–14, 27], where \( f_{j+1} \) is determined from performance criterion (3) using measurements \( e_j \) and \( f_j \), and a model of \( J \). In the following corollary, standard norm-optimal ILC is recovered as a special case of the framework in the present paper.
Connection 1 (Recovering standard norm-optimal ILC)  
Given a model of $J$ denoted as $\hat{J}$, set $\varepsilon_j = 1$ and $B_j = (\hat{j}^TW_c\hat{j} + W_f + W_{\Delta f})^{-1}$ in Algorithm 10, then

$$f_{j+1}^{po} = (\hat{j}^TW_c\hat{j} + W_f + W_{\Delta f})^{-1}(\hat{j}^TW_c\hat{j} + W_{\Delta f}) f_j + (\hat{j}^TW_c\hat{j} + W_f + W_{\Delta f})^{-1}\hat{j}^TW_ce_j.$$  

Clearly, $f_{j+1}^{po}$ depends on both measurement data and a model, hence robustness of the convergence and performance properties is a vital issue. Indeed, this is evidenced by the numerous developments in robust norm-optimal ILC, see e.g., [21–24] and this will also be illustrated with experiments in the next section.

In [50], a parameter-optimal ILC algorithm is proposed. The following corollary recovers a general parameter-optimal ILC algorithm as a special case of the framework in the present paper.

Connection 2 (Recovering parameter-optimal ILC)  
Given a model of $J$ denoted as $\hat{J}$, set $B_j = I$ in Algorithm 10, then the resulting ILC algorithm is given by

$$f_{j+1}^{po} = (I - \varepsilon_j^{po}W_f) f_j + \varepsilon_j^{po} \hat{j}^TW_c\hat{j},$$  

where learning gain $\varepsilon_j^{po}$ follows by replacing $J$ with the model $\hat{J}$ in (25) and is given by

$$\varepsilon_j^{po} = \frac{||\hat{j}\hat{w}_j||^2}{||\hat{j}\hat{w}_j||^2_W + ||\hat{j}\hat{w}_j||^2_W + ||\hat{j}\hat{w}_j||^2_{\Delta f}}$$  

with $\hat{w}_j = \hat{j}^TW_c\hat{j} - W_f f_j$. In case $W_c = I$ and $W_f = W_{\Delta f} = 0$ the learning gain reduces to

$$\varepsilon_j^{po,w} = \frac{||\hat{j}\hat{e}_j||^2}{||\hat{j}\hat{e}_j||^2}$$

Essentially, algorithm (27) is an adjoint-type ILC algorithm with a trial-varying learning gain $\varepsilon_j^{po}$ that follows using (25). As is also mentioned for norm-optimal ILC, algorithm (27) is model-based and robustness against model uncertainties is again a key issue, see e.g., [16] and [17] where robustness of algorithm (27) is investigated. In then next section, the main experimental results are presented.

6. EXPERIMENTAL VALIDATION

In this section, the proposed data-driven algorithms are experimentally validated and compared with model-based approaches norm-optimal ILC and parameter-optimal ILC constituting contribution 5, see Section 1. The system used in the experiments is an industrial flatbed printer, with multiple in- and outputs. The results demonstrate both enhanced performance as well as enhanced robustness properties. In particular, the following five approaches are compared

1. proposed data-driven quasi-Newton ILC Algorithm 10 with Hessian estimation in Procedure 12, see Summary 13 for an overview,

2. proposed data-driven quasi-Newton ILC Algorithm 10 with a fixed Hessian $B_j := I$ and data-driven optimal step size $\varepsilon_j$, see (25) in Procedure 12,

3. proposed data-driven gradient-descent ILC Algorithm 2 with a fixed step size $\varepsilon$ using Procedure 4,

4. standard norm-optimal ILC, with a model-based Hessian and step size $\varepsilon_j = 1$, see Connection 1,
5. parameter-optimal ILC, with a fixed Hessian $B_j := I$ and model-based step size $\varepsilon_j$, see Connection 2.

Note that the difference between approach 2 and 3 is either a varying (optimal) learning gain $\varepsilon_j$ (approach 2), or a fixed learning gain $\varepsilon$ (approach 3). Note that several implementation aspects are addressed in the appendices. In particular, a method to improve the signal-to-noise ratio in Procedure 4 is presented in Appendix A.1. Secondly, an approach to deal with friction in motion systems is proposed and compared with a pre-existing approach in [35] in Appendix A.2.

The outline of the present section is as follows. First, the experiment system and experimental test case is elaborated on. Next, the specific settings of the validated ILC algorithms are presented, followed by the main experimental results.

6.1. System description

The system used in the experimental validations is an industrial flatbed printer. The flatbed printer is shown in Fig. 5 and an overview is presented in Fig. 6. The system has four degrees of freedom: the carriage has translations $s$ and $z$, the gantry has a translation $x$, and a rotation $\varphi$ which is defined around the point $p$ that is fixed to the center of the gantry. The coordinate $s$ is fixed to the translation direction of the carriage and rotates with $\varphi$, the coordinates $z$ and $x$ are absolute.

The gantry is considered for control in the present paper and is denoted as system $P$. The gantry is controlled in $x$ and $\varphi$ direction using force actuators $u^1$ and $u^2$, see Fig. 6. The system response $y = Pu$ with

$$y = \begin{bmatrix} x \\ \varphi \end{bmatrix}, \quad u = \begin{bmatrix} u^1 \\ u^2 \end{bmatrix},$$
the output and input, respectively. The system operates in closed loop with a given feedback controller $C$. The input $u = Ce + f$, where $f = [f_1 f_2]^T$ are the ILC control signals and $e = \hat{r} - y$ with $\hat{r}$ the reference for the feedback controller with

$$\hat{r} = \begin{bmatrix} r_x \\ r_\varphi \end{bmatrix},$$

where $r_x$ is the reference for $x$ and $r_\varphi$ is the reference for $\varphi$ and $e = [e_1 e_2]^T$ are the corresponding tracking errors. The error in closed-loop operation is given by $e = SF - Jf$, with $S := (I + PC)^{-1}$ and $J := (I + PC)^{-1}P$. The closed-loop system is captured in the control structure in Fig. 2 by introducing trial index $j$ and setting $r := SJ\hat{r}$, yielding $e_j = r - Jf_j$. Since $e_j$ is directly measured, it is not necessary to compute $r = SJ\hat{r}$. The dedicated gradient experiments in Procedure 4 can be performed in closed-loop by straightforwardly setting $\tilde{r} = 0$, since in this case $e_j = Jf_j$. An approach for nonzero $\tilde{r}$ with the purpose of reducing mechanical friction effects is developed in Appendix A.2.

### 6.2. Model identification for comparative purposes

The five approaches defined in Section 2 and Section 4 are compared, these include the proposed data-driven algorithms and pre-existing model based approaches. To enable implementation of the model-based approaches, a model of $J$ has been identified using frequency response measurements and parametric model identification. Note that these models are not required to implement the proposed data-driven algorithms.

The system is excited with Gaussian noise and the sampling frequency is 1 kHz. A Von Hann window is used to deal with leakage effects. The non-parametric model $J_{frf}(z)$ is estimated using the procedure in [51, Chapter 3]. The frequency response measurement $J_{frf}(z)$ results from 300 averaged measurement blocks to reduce the variance. The obtained frequency resolution is 0.5 Hz.
The parametric model \( \hat{J}(z) \) is estimated using an iterative identification procedure in [52]. The resulting underlying state-space model for \( \hat{J}(z) \) is of 44th order. The Bode diagrams of \( J_{\text{trf}}(z) \) and \( \hat{J}(z) \) are shown in Fig. 7. The identified model \( \hat{J}(z) \) corresponds well with the measurement \( J_{\text{trf}}(z) \) for frequencies up to approximately 175 Hz.

Further analysis of the Bode diagram reveals that strong interaction is present, i.e., the magnitudes of the off-diagonal entries in \( J_{\text{trf}}(z) \) are in the same order of magnitude as the diagonal entries. It is therefore expected that MIMO ILC can achieve a significant performance improvement for this system.

6.3. ILC design and test case

First the selection of weighting matrices in (3) is presented. This is followed by a detailed overview of the specific settings for the compared algorithms. Finally, the selection of reference \( r \) is elaborated on.

6.3.1. Weighting matrices

The weighting matrices in performance criterion (3) are selected identical for the five tested algorithms to enable a direct comparison of the cost function values. Specifically, \( W_e = I \cdot 10^6 \), \( W_f = I \cdot 10^{-8} \), and \( W_{\Delta f} = I \cdot 10^{-3} \), with \( I \) of appropriate dimensions. The main motivation for these settings is to reduce \( e_j \) as much as possible. The value for \( W_e \) is for scaling purposes since the values of \( e_j \) are in the order \( 10^{-6} \), while the values \( f_j \) are in the order of 1. The matrix \( W_f \) ensures that large control signals are penalized. In addition, \( \hat{J} \) is singular thus \( W_f \cdot \hat{J} + W_f > 0 \) as is required in Theorem 7. The value for \( W_f \) has been increased from extremely small values to a level where the control signals are acceptable for the flatbed printer. It is known that \( W_f \) attributes to robustness at the expense of larger tracking errors, see e.g., [22]. In this test case, \( W_f \) is deliberately not increased further to demonstrate robustness issues with the model-based approaches. Finally, a relatively small \( W_{\Delta f} \) is introduced to attenuate trial-varying disturbances.

6.3.2. Algorithm settings

The proposed algorithms in Section 2.3 and Section 4.1 are experimentally validated and compared with the model-based special cases in Section 5. An overview of the algorithms with specific settings is presented in Table 1. Three variants of the proposed data-driven methods are tested and compared with the two model-based special cases in Section 5.

6.3.3. Reference trajectories

The reference trajectories \( r_x \) and \( r_{\phi} \) are presented in Fig. 8. The trial length \( N = 1500 \) samples with a sampling frequency \( f_s = 1 \) kHz. The reference for the \( x \)-direction is a fast forward and backward movement of the gantry with a displacement of 0.1 m. These step-wise trajectories are typical for printers and general motion systems, including pick-and-place systems. The backward movement leads to approximately identical initial conditions at the beginning of a trial. The reference \( r_{\phi} \) is zero, i.e., the goal is to keep the gantry rotation zero during movement in \( x \), which is also typical for printing. Note that although \( r_{\phi} \) is zero, the errors with feedback control are nonzero due to the interaction in \( J \), see Fig. 7 and the discussion in the previous section, hence this is a full multivariable control problem.

In the following section, the main experimental results are presented.

6.4. Main experimental Results

The experiments are invoked with \( f_0 = 0 \), hence the first trial hence corresponds to the performance of the system with feedback control only. For the data-driven approaches, a scaling \( \alpha_j \) is introduced to attenuate noise effects, see Appendix A.1, and chosen such that the absolute peak value of \( W_e e_j \) is 20% of the maximal allowable input range for \( J \), which is 10 V. In total, 100 trials are performed with each of the five approaches in Table 1. The results are presented in Fig. 9 and Fig. 10.

One of the key results in the present paper are the cost function values that are shown in Fig. 9. The results demonstrate significant performance improvements for the proposed data-driven approaches.
Table I. Overview algorithm settings

<table>
<thead>
<tr>
<th>Proposed data-driven approaches</th>
<th>Implementation</th>
<th>Learning update</th>
<th>Hessian</th>
<th>Learning gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) quasi-Newton algorithm (●)</td>
<td>Summary 13</td>
<td>see (23)</td>
<td>BFGS $H_0 = I$, see (24)</td>
<td>data-driven optimal $\epsilon_j$ in (25)</td>
</tr>
<tr>
<td>2) quasi-Newton algorithm without BFGS (□)</td>
<td>Summary 13</td>
<td>see (23)</td>
<td>$B_j = I \gamma_j$</td>
<td>data-driven optimal $\epsilon_j$ in (25)</td>
</tr>
<tr>
<td>3) gradient-descent (★)</td>
<td>Summary 13</td>
<td>see (6)</td>
<td>$B_j = I \gamma_j$</td>
<td>fixed $\epsilon = 0.5\epsilon = 1.18$ in (14)</td>
</tr>
</tbody>
</table>

| Pre-existing model-based approaches | | | | |
| 4) norm-optimal (●) | Connection 1 | see (26) | model-based, see Conn. 1 | - |
| 5) parameter-optimal (●) | Connection 2 | see (27) | - | model-based optimal $\epsilon_j$ in (28) |

Figure 8. References: $r_x$ (— solid black line) a 0.1 m smooth forward and backward step, $r_\varphi$ (— solid red line) zero rotation during the step.

Figure 9. Cost function values $J(f_j)$ for the five approaches in Table I: 1) (+) proposed data-driven quasi-Newton algorithm, 2) (□) proposed data-driven quasi-Newton algorithm without Hessian estimation, 3) (+) proposed gradient-descent, 4) (∆) standard norm-optimal ILC, 5) (○) parameter optimal ILC. The results demonstrate significant performance improvements for the proposed data-driven approaches and that both pre-existing model-based approaches have robustness issues.
and that both pre-existing model-based approaches have robustness issues. The following main observations are made.

- Proposed extended algorithm 1 with Hessian estimation achieves approximately a factor 10 higher performance after 100 trials than the other approaches. Approach 2 is identical to approach 1 without Hessian estimation. Between trials 0 – 20, very similar convergence behavior is observed. This is attributed to the initial Hessian estimate $B_0 = I$ in approach 1, hence similar behavior is expected for the first few trials until the Hessian estimate for approach 1 improves. Indeed, the convergence speed of approach 1 increases significantly after 20 trials leading to a much larger performance improvement. Approach 3 is the gradient-descent algorithm with a fixed step size. The difference with approach 2 lies in either a varying or a fixed learning gain, see Table I. Indeed as expected, this algorithm has the slowest convergence of the data-driven approaches, see the corresponding discussion in Section 3 and Section 4.

- Approach 4 is standard norm-optimal ILC. The results show a fast decrease of cost function for trials 0 – 3. At trial 4, oscillations start to appear in the tracking errors, which continue to grow until this behavior necessitated to stop the experiment after 10 trials. This behavior can be explained with the lack of robustness against modeling errors. Note that a high-fidelity model is used, see Fig. 7.

- Approach 5 is parameter-optimal ILC. The results show a slightly faster decrease of the cost function than the proposed gradient-descent approach 3 for trials 0 – 40. Still, the convergence is slower than approach 2, that uses a very closely related learning update, compare (23) with $B_j = I$ and (27). Indeed, a model is required to compute $\varepsilon_j$, see (28), in contrast to approach 1 and 2 where the stepsize $\varepsilon_j$ is determined using an additional dedicated experiment, see the discussion following Procedure 12. Modeling errors reduce the accuracy of the optimal $\varepsilon_j$ in (28) and could explain the decreased convergence rate. The results also show robustness issues for trials 50 – 80, which are indicated in Fig. 9. The cost function values appear to oscillate between trials 50 – 80, these oscillations eventually dissipate. This behavior could also be attributed to modeling errors in both the learning update, see (27), and (28).

In Fig. 10, time domain measurements of $e_j$ and $f_j$ for $j = 4$, $j = 9$ and $j = 99$ are shown. The results support the earlier conclusions since the proposed extended approach clearly achieves the smallest tracking errors after 100 trials. The other approaches have comparable tracking performance. Unstable behavior of standard norm-optimal ILC is prominently visible. In trial 4, oscillations appear in $e_1$ and at trial 9 these oscillations have significantly increased. Although difficult to observe in Fig. 10, close analysis of $f_1$ (black) and $f_2$ (black) reveals small oscillations in the control signals. Interestingly, the control signals of the proposed approach 1 in Table I for trial 99 (green) have a very similar shape as the control signals of norm-optimal ILC for trial 4, but without these small oscillations.

To conclude, the experimental results highlight performance and robustness improvements of the proposed methods with respect to the pre-existing approaches for the test-case on the Arizona flatbed printer.

7. CONCLUSION

In this paper a fully data-driven optimal ILC framework for multivariable systems is presented. The key idea is to replace the model in the learning update with dedicated experiments on the system. The main difficulty lies in the multivariable aspect. The need for a model is eliminated by measuring the gradient of a performance criterion using specially crafted intermediate experiments. To improve convergence speed, an extension is proposed that employs Hessian estimation. The efficacy of the developed framework is highlighted through comparative experiments with closely related model-based approaches. Both increased performance and robustness are
Figure 10. Time domain measurements $e_j$ and $f_j$ of the five approaches in Table I for trials 4 (left), 9 (center), and 99 (right): 1) (— solid green line) proposed data-driven quasi-Newton algorithm, 2) (— solid blue line) proposed data-driven quasi-Newton algorithm without Hessian estimation, 3) (— solid yellow line) proposed gradient-descent, 4) (— solid black line) standard norm-optimal ILC, 5) (— solid red line) parameter optimal ILC. The results show that the proposed data-driven quasi-Newton approach clearly achieves the smallest tracking errors after 99 trials, standard norm-optimal ILC shows fast convergence in 4 trials but also increasing oscillations in the tracking error that eventually necessitated to stop the experiment after 9 trials.

demonstrated on an industrial multivariable flatbed printer. It is expected that the results will replicate well in other applications, as the experiments are successful while the printer suffers from disturbances and nonlinearities including friction, noise, cable slab effects, and temperature-dependent actuator forces.

The key advantage of the presented framework is the fact that no model is required, which comes at the cost of additional experiment time that grows bi-linearly with the number of inputs and outputs of the system. Hence, the approach is particularly useful in cases of dealing with a large modeling effort and inexpensive experiments, such as in mechatronic printing systems. In case model knowledge is already available, the convergence speed can be enhanced further by using a model in the initial Hessian estimate.
Ongoing research is towards extending the approach to a sampled-data setting, e.g., using lines in [53], and investigating the effects of measurement noise and trial-varying disturbances. A first step towards addressing such effects in case they are severe could be including basis functions, see e.g., [5] and [48], for increased convergence speed, increased robustness against measurement noise, and introducing extrapolation capabilities with respect to the reference.

ACKNOWLEDGEMENT

The authors would also like to acknowledge the fruitful discussions with Maarten Steinbuch and Sjirk Koekebakker. Robbert Voorhoeve is acknowledged for identification of the parametric model for the comparative experiments.

This work is supported by Océ Technologies, P.O. Box 101, 5900 MA Venlo, The Netherlands. This work is also supported by the Innovational Research Incentives Scheme under the VENI grant “Precision Motion: Beyond the Nanometer” (no. 13073) awarded by NWO (The Netherlands Organisation for Scientific Research) and STW (Dutch Science Foundation).

REFERENCES


21. Son T, Pipeleers G, Swevers J. Experimental Validation of Robust Iterative Learning Control on an Overhead Crane Test Setup. *IFAC Triennial World Congress, Cape Town, South Africa* 2014; :5981–5986.


The implementation of Procedure 5 can benefit from addressing several implementation aspects. All measured signals contain noise. Consequently, the effects of noise in the dedicated adjoint experiments in step 2 of Procedure 5 (and also the identical step 2 in Summary 13) may affect the performance and convergence of the proposed ILC algorithms. Therefore, an approach to improve the signal-to-noise ratio is developed in this section.

A. IMPLEMENTATION ASPECTS

A.1. improving signal-to-noise ratio

The implementation of Procedure 5 can benefit from addressing several implementation aspects. All measured signals contain noise. Consequently, the effects of noise in the dedicated adjoint experiments in step 2 of Procedure 5 (and also the identical step 2 in Summary 13) may affect the performance and convergence of the proposed ILC algorithms. Therefore, an approach to improve the signal-to-noise ratio is developed in this section.
Consider the dedicated gradient experiment in Procedure 4 that is used in step 2 of Procedure 5 and Summary 13. The result of step 2 is given by $v_j = J^T W e_j$. Suppose that additive measurement noise $\eta_j$ is present on each measured signal in Procedure 4, then define the resulting measurement as $v^\eta_j = v_j + \eta_j$, which is given by

$$v^\eta_j = \eta_j + J^T W e_j.$$ 

The main idea is to attenuate the effect of $\eta_j$ by scaling $e_j$ with a factor $\alpha_j \in \mathbb{R}$ prior to Procedure 4 and compensating that scaling afterwards. Following this approach, $v^\eta_j$ is recast to

$$v^\eta_j = \frac{1}{\alpha_j} \left( \eta_j + J^T W e_j e_j \right) = \frac{\eta_j}{\alpha_j} + J^T W e_j.$$ 

Clearly, the effect of $\eta_j$ in $v^\eta_j$ can be made arbitrarily small by increasing $\alpha_j$. In practice, $\alpha_j$ can be used to trade-off signal-to-noise ratio in the data-driven learning procedure versus allowable input range and nonlinearities such as sensor quantization in the experimental system. A good value for $\alpha_j$ is hence highly application-specific and a suitable $\alpha_j$ can be obtained from system knowledge or appropriate pre-testing.

### A.2. Dealing with friction in motion systems

During the experiments, it is experienced that mechanical friction in motion systems influences the results of Procedure 4. A method to deal with friction is proposed in this section and compared with a pre-existing approach to deal with friction in [35]. The increased effectiveness of the proposed approach is illustrated with experiments and simulations.

The dedicated gradient experiments in Procedure 4 can be directly applied to the closed-loop system in Section 6.1 if $\tilde{r} = 0$ during the gradient experiments, as discussed in Section 6.1. In this case, the gantry of the Arziona flatbed printer, see Fig. 5 and Fig. 6, is at standstill and the presence of mechanical friction negatively affects the results of the dedicated gradient experiments, as will be demonstrated.

The proposed approach to mitigate the effects of friction is to use a very small constant velocity $\tilde{r}$ during the gradient experiments in Procedure 4, instead of $\tilde{r} = 0$ as required from the closed-loop setting described in Section 6.1. In this case, a perturbation is present on the result $v_j$ in step 2 in Procedure 5 and Summary 13. The perturbed result is given by

$$v^r_j \neq 0 = v_j - r$$

with $r = S \tilde{r}$ and $S = (I + PC)^{-1}$ as earlier given in Section 6.1. Clearly, in case $r = 0$, $v^r_j \neq 0 = v_j$, and step 2 remains identical to the open-loop case. Therefore, the impact of the additional term $\tilde{r}$ should be as small as possible, since it directly introduces an error in the gradient experiment.
Two measures to reduce the effect of nonzero references are used. Firstly, $C$ and $\tilde{r}$ are designed such that $S\tilde{r}$ is close to zero for constant velocity references. Secondly, the scaling approach proposed in Section A.1 also attenuates the effect of $r$ in $v'_j$ in exactly the same fashion as the measurement noise.

A pre-existing approach to deal with friction in data-driven ILC experiments for SISO systems is presented in [35, Section 6]. In the latter, $\tilde{r} = 0$ and a sign change in $e_j$ is applied every other trial before using the data-driven gradient approach, and changed back afterwards. The rationale behind this method is that the error in the gradient due to friction averages out as sufficiently many trials are performed. This method can be used in the MIMO framework of the present paper by applying these sign changes to $e_j$ in step 2 of Procedure 4.

Gradient-descent ILC algorithm 3 in Table I is used to validate these methods. In total, 40 trials are performed using the reference and ILC settings of the previous section. The results are presented in Fig. 11 and Fig. 12.

The cost function values are shown in Fig. 11. It shows that the proposed constant velocity approach performs significantly better than the pre-existing approach. The results also show that ignoring friction leads to a non-converging implementation since the cost function value start to increase after 20 trials. The cost function values for the pre-existing approach exhibit an alternating behavior, which is also observed in [35, Figure 17].

Time domain measurements and simulations of $v_0$, see step 2 in Summary 13, are shown in Fig. 12. It shows that the measurements correspond well with the simulation for the proposed method, and that the pre-existing approach and ignoring friction lead to large discrepancies. Given the close correspondence of the model with the frequency response measurements in Fig. 7, it is expected that $v_0$ for the proposed method is also closer to the true gradient than the other approaches.