

## Cross-coupled iterative learning control

### A computationally efficient approach applied to an industrial flatbed printer

Aarnoudse, Leontine; Kon, Johan; Classens, Koen; van Meer, Max; Poot, Maurice; Tacx, Paul; Strijbosch, Nard; Oomen, Tom

**DOI**

[10.1016/j.mechatronics.2024.103170](https://doi.org/10.1016/j.mechatronics.2024.103170)

**Publication date**

2024

**Document Version**

Final published version

**Published in**

Mechatronics

**Citation (APA)**

Aarnoudse, L., Kon, J., Classens, K., van Meer, M., Poot, M., Tacx, P., Strijbosch, N., & Oomen, T. (2024). Cross-coupled iterative learning control: A computationally efficient approach applied to an industrial flatbed printer. *Mechatronics*, 99, Article 103170. <https://doi.org/10.1016/j.mechatronics.2024.103170>

**Important note**

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

**Copyright**

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

**Takedown policy**

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



# Cross-coupled iterative learning control: A computationally efficient approach applied to an industrial flatbed printer<sup>☆,☆☆</sup>

Leontine Aarnoudse<sup>a,\*</sup>, Johan Kon<sup>a</sup>, Koen Classens<sup>a</sup>, Max van Meer<sup>a</sup>, Maurice Poot<sup>a</sup>, Paul Tacx<sup>a</sup>, Nard Strijbosch<sup>a</sup>, Tom Oomen<sup>a,b</sup>

<sup>a</sup> Department of Mechanical Engineering, Control Systems Technology, Eindhoven University of Technology, Eindhoven, The Netherlands

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

## ARTICLE INFO

### Keywords:

Iterative learning control  
Feedforward control  
Contour tracking

## ABSTRACT

Cross-coupled iterative learning control (ILC) can improve the contour tracking performance of manufacturing systems significantly. This paper aims to develop a framework for norm-optimal cross-coupled ILC that enables intuitive tuning of time- and iteration-varying weights of the exact contour error and its tangential counterpart. This leads to an iteration-varying ILC algorithm for which convergence conditions are developed. In addition, a resource-efficient implementation is developed that reduces the computational load significantly and enables the use of long reference signals. The approach is experimentally validated on an industrial flatbed printer.

## 1. Introduction

Ever-increasing demands for accuracy and speed in manufacturing require improved and problem-specific control strategies that focus on the actual performance requirement, such as cross-coupled control for contour tracking. Following a contour accurately is essential for the product quality in many multiple-input multiple-output (MIMO) applications such as printing, CNC machining and additive manufacturing. Standard control approaches that aim at following time-based references are not suitable for these applications, especially for increasingly complex parts. Complex structures such as curved surfaces and sharp curvature variations, manufactured at high feedrates, often lead to high contour errors [1]. To avoid increased errors and retain high product quality and efficiency, dedicated control strategies need to be developed.

Control strategies aimed at minimizing contour errors range from offline to online methods, including trajectory generation and precompensation [2], improved control of the individual axes, and cross-coupled feedback control [3–5]. These methods suffer from several limitations. Offline trajectory generation and precompensation is fully model-based, and the performance depends completely on model accuracy. Improved control of the individual axes does not directly take into account the contour error, and cross-coupled feedback control leads to a linear time-varying (LTV) system even if the original MIMO system is linear time-invariant (LTI), thus increasing the complexity of the

feedback control significantly. In addition, online computation times as well as stability requirements limit the accuracy of the contour error approximation.

Instead of online cross-coupled feedback control, norm-optimal cross-coupled iterative learning control is considered. Iterative learning control (ILC) is a control approach that uses measured data in conjunction with model knowledge to iteratively design feedforward or reference signals. This leads to high performance after only a small number of experiments. In norm-optimal ILC [6,7], a cost function is minimized that weights the error, i.e., the time-based deviation from the reference for each individual axis, as well as the input signal. This leads to high performance for the individual axes. However, standard norm-optimal ILC has limitations for contour tracking applications, especially if the input signals are restricted, for example due to actuator saturation. ILC often leads to extremely large inputs in sharp corners, where it may be preferable to reduce the speed instead, and to limited velocities in straight parts due to the time-based reference. Using cross coupling to include the contour error in norm-optimal ILC can lead to a significant increase in performance.

The performance of cross-coupled ILC depends on three factors. First, high accuracy of the contour error estimate is essential for accurate tracking. For cross-coupled feedback control, several online approximation approaches exist that vary in accuracy and computational load, ranging from linear and circular [3] to more complex parameter-based approximations [8]. In ILC, the input signal is computed offline

<sup>☆</sup> This work is part of the research programme VIDI with project number 15698, which is (partly) financed by the NWO.

<sup>☆☆</sup> This paper was recommended for publication by Associate Editor Takenori Atsumi.

\* Corresponding author.

E-mail address: [l.i.m.aarnoudse@tue.nl](mailto:l.i.m.aarnoudse@tue.nl) (L. Aarnoudse).

and therefore, the exact contour error can be used instead of this type of approximations. The exact contour error can be constructed from the individual axes errors through time- and iteration-varying coupling gains. While time-varying components are easily incorporated in norm-optimal ILC, this also requires an ILC framework that allows for variation over iterations.

Second, the design of cross-coupled ILC should include a clear and intuitive trade-off between accuracy and speed. Accuracy and speed are represented by respectively the contour error and the tangential error that is perpendicular to the contour error. Including a weight on this second error enforces the system to move over the contour. Since the relative importance of these errors often varies over the reference, e.g., in corners the contour error is essential, whereas on straight parts the speed can be increased to reduce the tangential error, the framework should enable time-varying weighting.

Third, implementation aspects are a major requirement, in particular for MIMO systems. Existing cross-coupled ILC methods are limited to references with a relatively small number of samples. This is because the ILC update law employs lifted system matrices, the size of which depends on the reference length, and involves inversion of these matrices. The computational load of these inversions scales badly with the signal length [9]. For cross-coupled ILC, a low-order solution that avoids inversion of large matrices and allows for long reference signals is essential.

In [10], an approach to cross-coupled ILC is presented that combines standard ILC and feedback control for the individual axes with PD-based ILC for an approximation of the contour error. The method is extended to norm-optimal cross-coupled ILC in [11], in which time-varying weighting is used. Both approaches use linear contour error estimates, which limits the achievable performance. In addition, while the norm-optimal cross-coupled ILC approach weights both the individual axes and the contour error, the tuning is not intuitive since the trade-off between speed and accuracy, i.e., tangential and contour errors, is not made explicit. The implementation in [11] is similar to that of standard norm-optimal ILC and requires the inversion of large matrices, thus limiting the approach to short reference signals. In [12], a non-lifted cross-coupled ILC framework is presented that is derived from the non-lifted norm-optimal ILC framework in [13]. This approach uses linear contour error approximations in individual cost functions at each error sample, leading to a PD-like ILC controller that does not allow for a direct feedthrough term in the plant. The cross-coupled ILC approach introduced in [14] uses an equivalent contour error that approximates the actual contour error provided that it is small, and does not enable tuning of the trade-off between contour and tangential errors.

Although significant steps have been taken to improve contour tracking through cross-coupled ILC, a framework that uses exact contour errors, enables intuitive tuning for both speed and accuracy, and which can be implemented efficiently for any length of reference signal is lacking. This paper aims to address these aspects as follows.

- A cost function is introduced that enables using exact contour errors with intuitive tuning of the time- and iteration-varying weights.
- Conditions for monotonic convergence of the cross-coupled ILC algorithm are developed.
- A resource-efficient implementation is presented that interprets the ILC update law as a linear quadratic tracking problem, enabling fast computations for iteration-varying weighting or coupling matrices and long reference signals.
- The approach is experimentally validated on an industrial flatbed printer.

Early and preliminary theoretical results appeared in [15]. The present paper extends these results with details on the implementation and the design, and by an extensive experimental case study on an industrial flatbed printer.

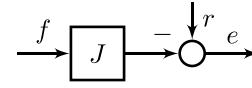


Fig. 1. Generic control scheme with system  $J$ , input  $f$ , disturbance  $r$  and error  $e$ .

The paper is constructed as follows. In Section 2, the problem is introduced. Cross-coupled ILC is introduced in 3, and convergence conditions are developed in 4. In Section 5, a resource-efficient implementation is presented. The choice of design parameters is elaborated upon in Section 6. Experimental results are presented in Section 7 and finally, conclusions are given in Section 8.

## 2. Problem formulation

In this section, first the contour tracking problem is defined and secondly, the norm-optimal ILC framework is introduced.

### 2.1. Contour tracking

Consider a discrete-time, linear time-varying (LTV) system  $J$  with  $n_i$  inputs and  $n_o$  outputs, with error  $e$  given by

$$e = y_d - y = r - Jf, \quad (1)$$

where  $y_d$  and  $y$  denote respectively the reference and the system output,  $f$  is an input signal and  $r$  encompasses all exogenous disturbances. The system  $J$  and input  $f$  are given in lifted form by

$$J = \begin{bmatrix} H^{0,0} & \dots & 0 \\ \vdots & \ddots & \vdots \\ H^{N-1,0} & \dots & H^{N-1,N-1} \end{bmatrix}, \quad f = \begin{bmatrix} f(0) \\ f(1) \\ \vdots \\ f(N-1) \end{bmatrix},$$

with  $J$  the convolution matrix of the LTV system, which is a lower triangular matrix with a block Toeplitz structure and entries  $H^{i,j} \in \mathbb{R}^{n_o \times n_i}$ . The exact configuration of  $J$  depends on where  $f$  is injected in the open- or closed-loop system, as is elaborated upon in Section 6. The error  $e \in \mathbb{R}^{Nn_o}$  is written similar to input  $f \in \mathbb{R}^{Nn_i}$ , and  $e(k) \in \mathbb{R}^{n_o}$  and  $f(k) \in \mathbb{R}^{n_i}$  are of the form

$$e(k) = [e^1(k) \quad e^2(k) \quad \dots \quad e^{n_o}(k)]^T \\ f(k) = [f^1(k) \quad f^2(k) \quad \dots \quad f^{n_i}(k)]^T. \quad (2)$$

The aim of the system is to track a contour described by the reference  $y_d(k) \in \mathbb{R}^{n_o}$ . Consider a standard closed-loop system with system  $P(q)$  and controller  $C(q)$ , where  $q$  denotes the shift operator, and no additional disturbances. Then the disturbance  $r(k)$  at discrete-time index  $k$  is related to the reference  $y_d(k)$  through

$$r(k) = (I + P(q)C(q))^{-1} y_d(k). \quad (3)$$

The reference  $y_d(k)$  and the resulting error  $e(k) \in \mathbb{R}^{n_o}$  are a function of time, yet the goal is to track the contour described by  $y_d$  accurately in space rather than in time. To that end, the contour error  $\varepsilon_c(k) \in \mathbb{R}$  is defined as the distance between the position output  $y(k) \in \mathbb{R}^{n_o}$  and the closest point on the contour, as illustrated in Fig. 2.

The aim is to reduce the contour error, yet this error is not measured directly. Instead, the contour error is reconstructed from the measured individual axis errors through coupling gains. Each contour error sample  $\varepsilon_c(k) \in \mathbb{R}$  can be expressed as a function of  $e(k) \in \mathbb{R}^{n_o}$  through a vector of coupling gains  $c(k) \in \mathbb{R}^{n_o}$  according to

$$\varepsilon_c(k) = c(k)^T e(k). \quad (4)$$

In addition, a tangential error  $\varepsilon_t$  that is perpendicular to the contour error can be defined. This error is expressed similar to  $\varepsilon_c$  as a function of  $e(k) \in \mathbb{R}^{n_o}$ , with a different vector of coupling gains. The coupling gains depend on whether the exact contour error  $\varepsilon_c(k)$  or one of the various approximations  $\hat{\varepsilon}_c$  is used, as is further explained in Section 3.

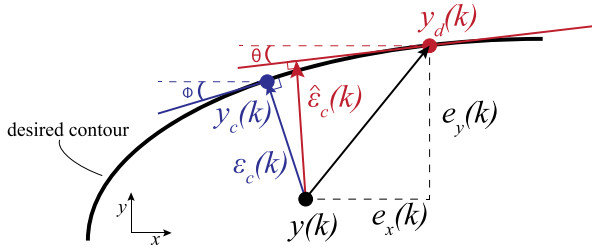


Fig. 2. For a 2D-system with reference  $y_d(k)$ , position output  $y(k)$  and individual axes errors  $e_x(k)$  and  $e_y(k)$ , the actual contour error  $\varepsilon_c(k)$  based on the closest point on the contour  $y_c(k)$  differs from linear approximation  $\hat{\varepsilon}_c(k)$ .

## 2.2. Norm-optimal iterative learning control

In norm-optimal iterative learning control (ILC), the input signal  $f$  of a system of the form (1) is updated iteratively, according to

$$e_j = r - Jf_j, \quad (5)$$

$$f_{j+1} = Qf_j + Le_j, \quad (6)$$

where error  $e_j$  and  $f_j$  depend on iteration  $j$ . In norm-optimal ILC, the update matrices  $Q$  and  $L$  follow from minimizing the cost function

$$\mathcal{J}(f_{j+1}) = \|e_{j+1}\|_{W_e}^2 + \|f_{j+1}\|_{W_f}^2 + \|f_{j+1} - f_j\|_{W_{\Delta f}}^2, \quad (7)$$

where the weighted 2-norm is given by  $\|e_{j+1}\|_{W_e}^2 = e_{j+1}^T W_e e_{j+1}$ . The cost function weights the error, as well as the input signal, leading to a regularization term that provides robustness in case of model uncertainty, and the change in input, in order to average over iterations and reduce the influence of iteration-varying disturbances. The minimizer of  $\mathcal{J}(f_{j+1})$  is determined analytically and is given by

$$f_{j+1} = \underbrace{(J^T W_e J + W_f + W_{\Delta f})^{-1} (J^T W_e J + W_{\Delta f})}_{Q} f_j + \underbrace{(J^T W_e J + W_f + W_{\Delta f})^{-1} J^T W_e}_{L} e_j, \quad (8)$$

see, e.g., [6,16] for a derivation. If the convolution matrix  $J$  is non-singular, norm-optimal ILC with  $W_e > 0$ ,  $W_f, W_{\Delta f} \geq 0$  leads to monotonic convergence of the sequence of input signals  $\{f_j\}$  in the 2-norm, as is further illustrated in the proof of Theorem 4. If  $J$  is singular then  $W_f > 0$  ensures monotonic convergence.

## 2.3. Problem formulation

The aim of this paper is to develop a framework for norm-optimal cross-coupled ILC that achieves high performance for contour tracking applications. This requires a cost function that directly takes into account the exact contour error, and that enables intuitive, time-varying tuning. The resulting framework should lead to monotonic convergence of the ILC input signal  $f$ , and a resource-efficient implementation that is suitable for use with long references is required.

## 3. Cost function design for contour tracking

In this section, a cost function is introduced that allows for different configurations of cross-coupled ILC. First, a cost function framework is introduced that allows the use of exact contour errors with iteration-varying coupling matrices. Then, it is illustrated that the framework also enables intuitive time- and iteration-varying weighting.

### 3.1. Cost function framework

The cost function that is introduced for cross-coupled ILC is similar to the standard norm-optimal ILC cost function in (7), including similar weighted 2-norms of the predicted error  $e_{j+1}$ , the input signal  $f_{j+1}$  and the change in input  $f_{j+1} - f_j$ . In contrast to standard norm-optimal ILC, the weighting matrices are now iteration-varying, and include coupling matrices to compute the contour error based on the measured error of the individual axes. The main reason for including iteration-varying weighting matrices is that the contour error, an iteration-invariant performance specification, depends on the measured individual axis errors through coupling matrices that depend on the error and vary over iterations. The cost function for cross-coupled ILC is given in generic form by

$$\mathcal{J}(f_{j+1}) = \|e_{j+1}\|_{W_{ec,j}}^2 + \|f_{j+1}\|_{W_{fc,j}}^2 + \|f_{j+1} - f_j\|_{W_{\Delta f,j}}^2. \quad (9)$$

The weighting-coupling matrices  $W_{ec,j}$ ,  $W_{fc,j}$  and  $W_{\Delta f,j}$  may be iteration-varying, as indicated by the subscript  $j$ . These weighting-coupling matrices can weight not only the errors and inputs of the individual axes, but also combinations of these errors such as different approximations of the contour error and the tangential error that is perpendicular to the contour error. The matrices are designed as

$$W_{ec,j} = C_{e,j}^T W_{e,j} C_{e,j} \quad (10)$$

$$W_{fc,j} = C_{f,j}^T W_{f,j} C_{f,j} \quad (11)$$

$$W_{\Delta f,j} = C_{f,j}^T W_{\Delta f,j} C_{f,j} \quad (12)$$

where different coupling matrices  $C_{e,j}$  and  $C_{f,j}$  are used the error and input. This allows for situations where one wants to minimize the contour error while also limiting the inputs to the individual axes. The coupling matrices are block diagonal matrices that are constructed as

$$C_{e,j} = \begin{bmatrix} C_{e,j}^1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & C_{e,j}^N \end{bmatrix}. \quad (13)$$

The blocks  $C_{e,j}^k \in \mathbb{R}^{n_{ce} \times n_o}$ , with  $n_{ce}$  the number of considered error components, have full column rank and describe the coupling of the axes at each sample. For a 2D system with  $e(k) = [e_y(k) \ e_x(k)]^T$ , the coupling matrices  $C_{e,j}^k$  that give the exact contour and tangential errors are given by

$$C_{e,j}^k = \begin{bmatrix} \cos(\phi(k,j)) & -\sin(\phi(k,j)) \\ \sin(\phi(k,j)) & \cos(\phi(k,j)) \end{bmatrix}, \quad (14)$$

where  $\phi(k,j)$  is the angle between the  $x$ -axis and the vector perpendicular to the contour error vector. The contour error vector is the vector from the position  $y_j(k)$  to the closest point on the contour  $y_{c,j}(k)$ , found through evaluation of the distance from  $y_j(k)$  to each point on contour  $y_d$  according to Algorithm 1, see also Fig. 2.

### Algorithm 1 Computing the contour error

```

1: for  $k = 1 : N$ 
2:   for  $l = 1 : N$ 
3:     Compute  $d(l) = \|y_d(l) - y_j(k)\|_2$ .
4:   end
5:   Find index  $l^* = \arg \min d(l)$ .
6:   The point closest to  $y_j(k)$  is  $y_{c,j}(k) = y_d(l^*)$ .
7:   The contour error is  $\varepsilon_{c,j}(k) = y_{c,j}(k) - y_j(k)$ .
8: end

```

Using (14) ensures that

$$C_{e,j}^k \begin{bmatrix} e_y(k) \\ e_x(k) \end{bmatrix} = \begin{bmatrix} \varepsilon_c(k) \\ \varepsilon_t(k) \end{bmatrix}. \quad (15)$$

Note that using the exact contour error in (9) may lead to coupling gains in  $C_{e,j}^k$  that vary over iterations. As a result, the weighting matrices and the cost function are iteration-varying as well.

The matrix  $C_{f,j}$  is designed similar to  $C_{e,j}$  and consists of blocks  $C_{f,j}^k \in \mathbb{R}^{n_{ef} \times n_o}$ . Since the MIMO system is not necessarily square, and different couplings for the input and error might be desired, the matrices  $C_{f,j}$  and  $C_{e,j}$  are not necessarily identical.

### 3.2. Time- and iteration-varying weighting

The structure of cost function (9) leaves a lot of freedom in the design of the weighting matrices  $W_{e,j}$ ,  $W_{f,j}$  and  $W_{df,j}$ . The matrices have the following block-diagonal structure:

$$W_{e,j} = \begin{bmatrix} W_{e,j}^1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & W_{e,j}^N \end{bmatrix}, \quad (16)$$

where the size of  $W_{e,j}^k \in \mathbb{R}^{n_{ee} \times n_{ee}}$  depends on that of the corresponding block  $C_{e,j}^k$  of the coupling matrix. Each block  $W_{e,j}^k$  is diagonal and applies weights to each element of the coupled error. For example, if

$$C_{e,j}^k = \begin{bmatrix} \cos(\phi(k, j)) & -\sin(\phi(k, j)) \\ \sin(\phi(k, j)) & \cos(\phi(k, j)) \end{bmatrix}, \quad (17)$$

such that

$$C_{e,j}^k \begin{bmatrix} e_y(k) \\ e_x(k) \end{bmatrix} = \begin{bmatrix} \varepsilon_c(k) \\ \varepsilon_t(k) \end{bmatrix}, \quad (18)$$

the first element on the diagonal of  $W_{e,j}^k$  puts a weight on the contour error  $\varepsilon_c(k)$ , and the second diagonal element puts a weight on the tangential error  $\varepsilon_t(k)$ . The design of  $W_{e,j}$  leaves room for time-varying weighting within one iteration, as well as iteration-varying weights, i.e., the blocks  $W_{e,j}^k$  need not be identical for all  $k$  or  $j$ . This is further illustrated in Section 6.

### 4. Convergence of iteration-varying ILC

The cost function with exact contour errors introduced in Section 3 leads to an iteration-varying ILC update due to the iteration-varying coupling matrices that are needed to compute  $\varepsilon_c$  exactly. In this section, conditions are developed for the monotonic convergence towards a closed 2-norm ball of cross-coupled ILC. In particular, it is shown that under mild conditions the use of iteration-varying weighting matrices as in (9) does not render the ILC system unstable, and convergence can still be shown. The iteration-varying ILC update based on cost function (9) is given by

$$f_{j+1} = Q_j f_j + L_j e_j, \text{ with} \quad (19)$$

$$Q_j = (J^T W_{ec,j} J + W_{ef,j} + W_{dfc,j})^{-1} (J^T W_{ec,j} J + W_{dfc,j}), \quad (20)$$

$$L_j = (J^T W_{ec,j} J + W_{ef,j} + W_{dfc,j})^{-1} J^T W_{ec,j}. \quad (21)$$

To show convergence, define the sets containing all possible filters  $Q_j$  and  $L_j$  as  $\mathcal{Q}$  and  $\mathcal{L}$ , respectively. The reference and position are sampled with finite resolution, and it is assumed that the set of iteration-varying weights is also finite. As a result, the sets  $\mathcal{Q}$  and  $\mathcal{L}$  are finite. Next, monotonic convergence towards a closed 2-norm ball is defined.

**Definition 2 (Closed 2-Norm Ball).** The closed 2-norm ball  $B_2(c, d)$  with center  $c \in \mathbb{R}$  and radius  $d \in \mathbb{R}_{\geq 0}$  is defined as  $B_2(c, d) := \{x \in \mathbb{R} \mid \|x - c\|_2 \leq d\}$ .

**Definition 3 (Monotonic Convergence Towards a Closed 2-Norm Ball).** The sequence  $\{y_i\}$ ,  $y_i \in \mathbb{R}$  is said to converge monotonically in the 2-norm to the 2-norm ball  $B_2(c, d)$  if there exists  $\kappa \in [0, 1)$  such that for all  $i \in \mathbb{Z}_{\geq 0}$ ,

$$\|y_{i+1} - c\|_2 \leq \kappa \|y_i - c\|_2 \quad \text{if } y_i \notin B_2(c, d), \quad (22)$$

$$y_{j+1} \in B_2(c, d) \quad \text{if } y_i \in B_2(c, d). \quad (23)$$

These definitions are employed in the following convergence theorem for cross-coupled ILC.

**Theorem 4.** The sequence of inputs  $\{f_j\}$  that follows from update law (19), with iteration-varying filters  $Q_j \in \mathcal{Q}$  and  $L_j \in \mathcal{L}$  that minimize criterion (9) according to (21), is monotonically convergent towards a closed 2-norm ball if the coupling matrix  $C_{e,j}$  has full column rank,  $W_{e,j} > 0 \forall j$  and either,

- if  $J$  is non-singular,  $W_{f,j}, W_{df,j} \geq 0 \forall j$ , or,
- if  $J$  is singular,  $C_{f,j}$  has full column rank and  $W_{f,j} > 0, W_{df,j} \geq 0 \forall j$ .

The proof is based on the following auxiliary result.

**Lemma 5.** For iteration (19), the following two statements are equivalent:

1. The sequence of inputs  $\{f_j\}$  with fixed  $Q_j = \bar{Q} \in \mathcal{Q}$  and  $L_j = \bar{L} \in \mathcal{L}$  for all  $j$  is monotonically convergent in the 2-norm to a fixed point.
2. The sequence of inputs  $\{f_j\}$  with iteration-varying  $Q_j \in \mathcal{Q}$  and  $L_j \in \mathcal{L}$  is monotonically convergent in the 2-norm towards a closed 2-norm ball.

The complete proofs of both Theorem 4 and Lemma 5 are given in Appendix. If the weighting-coupling matrices are iteration-invariant, for example because an approximation of the contour error based on iteration-invariant coupling gains is used, Theorem 4 ensures monotonic convergence in the 2-norm to a fixed point instead. Theorem 4 reduces the design of cross-coupled ILC for monotonic convergence to choosing suitable weights and couplings, enabling intuitive design as illustrated in Section 6.

### 5. Resource-efficient implementation

The iteration-varying ILC update law (19) that follows from cost function (9) with exact contour errors involves the matrices  $Q_j \in \mathbb{R}^{N n_i \times N n_i}$  and  $L_j \in \mathbb{R}^{N n_i \times N n_o}$ . Computing these matrices involves the inversion of the matrix  $(J^T W_{ec,j} J + W_{ef,j} + W_{dfc,j})$ , which is  $\in \mathbb{R}^{N n_i \times N n_i}$ , at each iteration. This operation is computationally expensive and may not even be feasible for increasing reference lengths  $N$  [9]. Therefore, in this section the cross-coupled ILC update law (19) is rewritten as a linear quadratic tracking (LQT) problem, for which a resource-efficient solution is given that reduces the computational load significantly. In this approach the matrices  $Q_j$  and  $L_j$  are not calculated explicitly, avoiding the inversion that would otherwise limit the size of the matrices of the lifted system and thus also the length of the reference signals.

#### 5.1. Norm-optimal ILC as an LQT problem

The closed-loop system  $J$  that is written in lifted form in the previous sections, is now rewritten to a state-space description with time-varying system matrices  $A^k$ ,  $B^k$ ,  $C^k$  and  $D^k$ . Due to the block-diagonal structure of  $W_{ec,j}$ ,  $W_{f,j}$  and  $W_{dfc,j}$  the cost function (9) can be written as

$$\begin{aligned} \mathcal{J}(f_{j+1}) = & \sum_{k=1}^N \|e_{j+1}(k)\|_{W_{ec,j}^k}^2 + \|f_{j+1}(k)\|_{W_{fc,j}^N}^2 \\ & + \|f_{j+1}(k) - f_j(k)\|_{W_{dfc,j}^k}^2, \end{aligned} \quad (24)$$

where, consistent with the previous notation,  $W_{ec,j}^k = (C_{e,j}^k)^T W_{e,j}^k C_{e,j}^k$  etc. From (5) it follows that  $e_{j+1} = e_j - J(f_{j+1} - f_j)$ . In addition,  $\Delta f_{j+1}(k) = f_{j+1}(k) - f_j(k)$  and  $\Delta e_{j+1}(k) = e_{j+1}(k) - e_j(k)$  are defined. This leads to the following theorem that relates the optimal input  $f_{j+1}$  in (9) to the solution of a linear quadratic tracking problem.



**Theorem 6.** The optimal ILC input that minimizes (9) with  $W_{\Delta f_{c,j}} > 0$  is the solution to the linear quadratic tracking problem with cost function

$$J(\Delta f_{j+1}) = \sum_{k=1}^N \Delta f_{j+1}^T(k) \underbrace{W_{\Delta f_{c,j}}^k}_{R_j^k} \Delta f_{j+1}(k) + \underbrace{\begin{bmatrix} e_j(k) + \Delta e_{j+1}(k) \\ f_j(k) + \Delta f_{j+1}(k) \end{bmatrix}^T \begin{bmatrix} W_{ec,j}^k & 0 \\ 0 & W_{fc,j}^k \end{bmatrix} \begin{bmatrix} e_j(k) + \Delta e_{j+1}(k) \\ f_j(k) + \Delta f_{j+1}(k) \end{bmatrix}}_{S_j^k}, \quad (25)$$

subject to the dynamics

$$\begin{aligned} \Delta x_{j+1}(k+1) &= A^k \Delta x_{j+1}(k) + B^k \Delta f_{j+1}(k) \\ \Delta y_{j+1}(k) &= \begin{bmatrix} \Delta e_{j+1}(k) \\ \Delta f_{j+1}(k) \end{bmatrix} \\ &= \begin{bmatrix} C^k \\ 0 \end{bmatrix} \Delta x_{j+1}(k) + \begin{bmatrix} D^k \\ I \end{bmatrix} \Delta f_{j+1}(k). \end{aligned} \quad (26)$$

**Proof.** The proof follows from substituting  $e_{j+1} = e_j - J(f_{j+1} - f_j)$  and  $\Delta e_{j+1} = -J(f_{j+1} - f_j) = -J\Delta f_{j+1}$  in (24), which is equivalent to (9). Adding  $\Delta f_{j+1}$  as additional output allows for reframing of the problem as an LQT problem with a direct feedthrough term. Taking  $W_{\Delta f_{c,j}} > 0$  ensures that  $S_j^k \geq 0$ ,  $R_j^k > 0$  and concludes the proof.  $\square$

The solution in Theorem 6 is identical to that of the lifted ILC update law (19), in contrast to the non-lifted approach in [12] which minimizes an individual cost function at each sample and as such is fundamentally different. Note that compared to the convergence Theorems 4, 6 also requires  $W_{\Delta f_{c,j}} > 0$ . This is typically not a limitation, since in practical applications it is usually desired to take  $W_{\Delta f_{c,j}} > 0$  to limit the amplification of iteration-varying disturbances [17].

### 5.2. Solving the LQT problem for cross-coupled ILC

The approach to solving discrete-time LQT problems with  $S_j^k \geq 0$ ,  $R_j^k > 0$  is well-known and described in, e.g., [18, Section 4.4] or [19] for the general situation with direct feedthrough and noise terms. The approach consists of defining a Hamiltonian system, leading to a two-point boundary value problem, to which a sweep method is applied that involves solving part of the equations backwards in time. For the cross-coupled ILC problem this leads to the following solution. The exact implementation is further explained in Section 5.3. Define

$$\bar{C}^k = \begin{bmatrix} C^k \\ 0 \end{bmatrix}, \quad \bar{D}^k = \begin{bmatrix} D^k \\ I \end{bmatrix}, \quad r_j(k) = \begin{bmatrix} -e_j(k) \\ -f_j(k) \end{bmatrix}.$$

The optimal input is given by  $f_{j+1}(k) = f_j(k) + \Delta f_{j+1}(k)$  and

$$\begin{aligned} \Delta f_{j+1}(k) &= -\bar{R}_j^{-1}(k) \bar{G}_j(k+1) \Delta x_{j+1}(k) \\ &\quad + \bar{R}_j^{-1}(k) (\bar{D}^k)^T S_j^k r_j(k) + \bar{R}_j^{-1}(k) (B^k)^T v_j(k+1), \end{aligned} \quad (27)$$

with

$$\bar{R}_j(k) = R_j^k + (\bar{D}^k)^T S_j^k \bar{D}^k + (B^k)^T G_j(k+1) B^k \quad (28)$$

$$\bar{G}_j(k+1) = (B^k)^T G_j(k+1) A^k + (\bar{D}^k)^T (S_j^k)^T \bar{C}^k. \quad (29)$$

The terms  $v_j(k+1)$  and  $G_j(k+1)$  follow from solving the following equations backwards in time:

$$G_j(k) = (A^k)^T G_j(k+1) A^k + (\bar{C}^k)^T S_j^k \bar{C}^k - \quad (30)$$

$$((A^k)^T G_j(k+1) B^k + (\bar{C}^k)^T S_j^k \bar{D}^k) \bar{R}_j^{-1}(k) \bar{G}_j(k+1)$$

$$v_j(k) = -\left(\bar{G}_j^T(k+1) \bar{R}_j^{-1}(k) (\bar{D}^k)^T - (\bar{C}^k)^T\right) S_j^k r_j(k)$$

$$-\left(\bar{G}_j^T(k+1) \bar{R}_j^{-1}(k) (B^k)^T - (A^k)^T\right) v_j(k+1), \quad (31)$$

with  $x(0) = 0$  and boundary conditions

$$G_j(N) = (\bar{C}^N)^T S_j^N \bar{C}^N \quad (32)$$

$$v_j(N) = (\bar{C}^N)^T S_j^N (\bar{D}^N \Delta f_{j+1} - r_j(N)). \quad (33)$$

A similar low-order solution to ILC, which omits the explicit formulation of the ILC update law as an LQT problem, is also applied to the specific cases of ILC for intersample behavior in [20] and norm-optimal ILC in [9].

### 5.3. Solving the LQT problem for cross-coupled ILC: implementation

The implementation for solving the LQT problem for cross-coupled ILC is summarized in the following algorithm. All computations involve only small matrices for each time step, resulting in much faster calculations compared to the full lifted matrices used in (19).

#### Algorithm 7 Resource-efficient ILC update

- 1: Given the time-varying system matrices  $A^k, B^k, C^k, D^k$  and the weighting matrices  $R_j^k$  and  $S_j^k \forall k$ , compute  $G_j(k)$  backwards in time according to (30), starting from the boundary condition  $G_j(N)$  in (32).
- 2: Using  $G_j(k)$ , compute  $v_j(k)$  backwards in time according to (31), starting from the boundary condition  $v_j(N)$  in (33).
- 3: Compute  $\bar{R}_j(k)$  and  $\bar{G}_j(k)$  using  $G_j(k)$  according to (28) and (29).
- 4: Compute the change in input signal  $\Delta f_{j+1}(k)$  using  $\bar{R}_j(k)$ ,  $\bar{G}_j(k)$ ,  $v_j(k)$  and the system dynamics (26) according to (27).
- 5: The optimal input for iteration  $j+1$  is given by  $f_{j+1}(k) = f_j(k) + \Delta f_{j+1}(k)$ .

## 6. Design and recovering existing methods

In this section, two design aspects, i.e., the choice between parallel and serial ILC configuration and the selection of weights, are elaborated upon. In addition, it is shown how several pre-existing approaches are encompassed by the framework presented in this paper.

### 6.1. Design: parallel or serial ILC configurations

The system  $J$  in Fig. 1 can represent both parallel and serial ILC configurations. In parallel ILC,  $J_{\text{par}}$  results from the process sensitivity  $(I + P(q)C(q))^{-1}P(q)$  of the closed-loop system, and  $f$  is a feedforward signal that is injected between the controller and the plant as illustrated in Fig. 3. In serial ILC, the input  $f$  is added to the reference signal and  $J_{\text{ser}}$  results from the closed loop  $(I + P(q)C(q))^{-1}P(q)C(q)$ , as illustrated in Fig. 4.

In cross-coupled ILC, the performance variables for the offline ILC computations, i.e., contour and tangential errors, differ from the online feedback variables, i.e., the individual axes errors. This leads to an inferential ILC situation which in some cases can lead to internal stability problems, see [9]. Since the presented framework computes the contour errors from the individual axes errors and the coupling matrices have full column rank, it is not possible for one of the individual axes to become unstable if the ILC system is monotonically convergent. However, it is possible that the control actions of the feedback and feedforward controllers oppose each other, because they are based on respectively the individual axes errors w.r.t. the original reference and the contour error. To avoid this, a serial ILC configuration may be used. In this configuration, the ILC input signal is used to adapt the reference, which is then tracked using the feedback controller, such that no feedforward term is included that could oppose the feedback action.

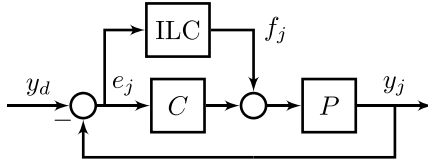


Fig. 3. Parallel ILC configuration where  $J_{\text{par}}$  follows from the transfer between  $f_j$  and  $e_j$ , given by  $(1 + PC)^{-1}P$ .

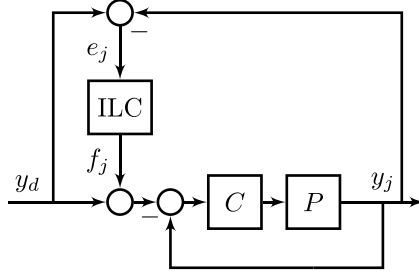


Fig. 4. Serial ILC configuration, where  $J_{\text{ser}}$  follows from the transfer between  $f_j$  and  $e_j$ , given by  $(I + PC)^{-1}PC$ .

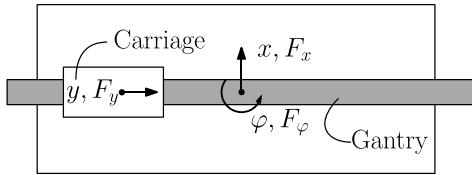


Fig. 5. Photograph (top) and schematic overview (bottom) of the Arizona flatbed printer.

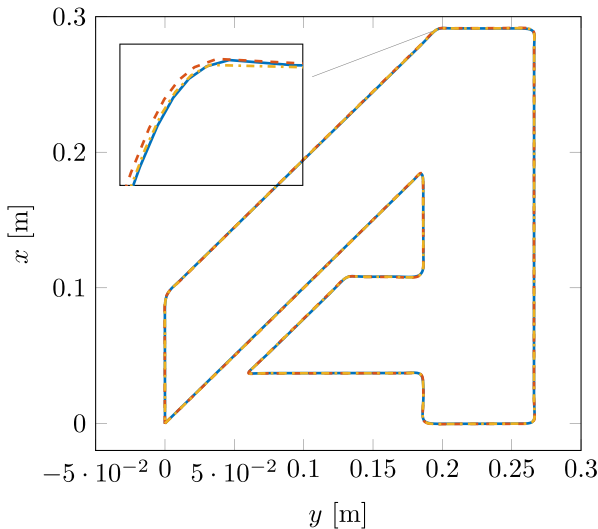


Fig. 6. Reference (—) and output without feedforward (---) and with cross-coupled ILC (-.-).

## 6.2. Design: selecting the weights

The weighting matrices in the cost function (9) can be time- and iteration-varying or invariant. The standard choice in norm-optimal ILC is to use invariant weighting matrices, often with scaled identity matrices  $W_e = I$ ,  $W_f = w_f I$  and  $W_{\Delta f} = w_{\Delta f} I$ . In this case  $w_f$  is often chosen as small as possible while still providing robustness against model uncertainty, and  $w_{\Delta f}$  is chosen non-zero to reduce the influence of iteration-varying disturbances.

In case of cross-coupled ILC, it is possible and often desirable to weight the contour error  $\epsilon_c$  heavier than the tangential error  $\epsilon_t$ . In addition, the ratio between these weights might vary over the trajectory, for example, in corners one might want to increase the weight on the contour error to increase accuracy, whereas on straight parts the weight on the tangential error could be increased to increase the speed and make up for lost time in the corners. To this end, time-varying weighting matrices can be used. An extensive case study of time-varying weight matrices for cross-coupled ILC with approximated contour errors is conducted in [11], the results of which can be applied directly to the cross-coupled ILC framework introduced here. Weighting approaches developed specifically for coupled systems with dissimilar dynamics, such as 3D printing systems [21], can also be incorporated directly. In addition to time-varying weights, (9) and Theorem 4 enable the use of iteration-varying weighting matrices. Iteration-varying weights can be used, for example, to initially choose  $W_{\Delta f}$  large to ensure fast reduction of the error, and reduce it later to average over iterations and reduce the influence of iteration-varying disturbances.

## 6.3. Recovering standard norm-optimal ILC and contour error approximations

The framework based on minimizing the cost function (9) encompasses existing approaches, such as standard norm-optimal ILC and cross-coupled ILC with linear approximations as used in [10]. These approaches are recovered through specific choices of the coupling matrices  $C_{e,j}^k$  in (13):

- Standard norm-optimal ILC:

$$C_{e,j}^k = I^{2 \times 2} \quad \forall k, j. \quad (34)$$

- Cross-coupled ILC with linear approximations of the contour error  $\hat{\epsilon}_c$  and individual axes errors:

$$C_{e,j}^k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \forall j, \quad (35)$$

where  $\theta(k)$  is the angle between the reference  $y_d(k)$  at sample  $k$  and the  $x$ -axis.

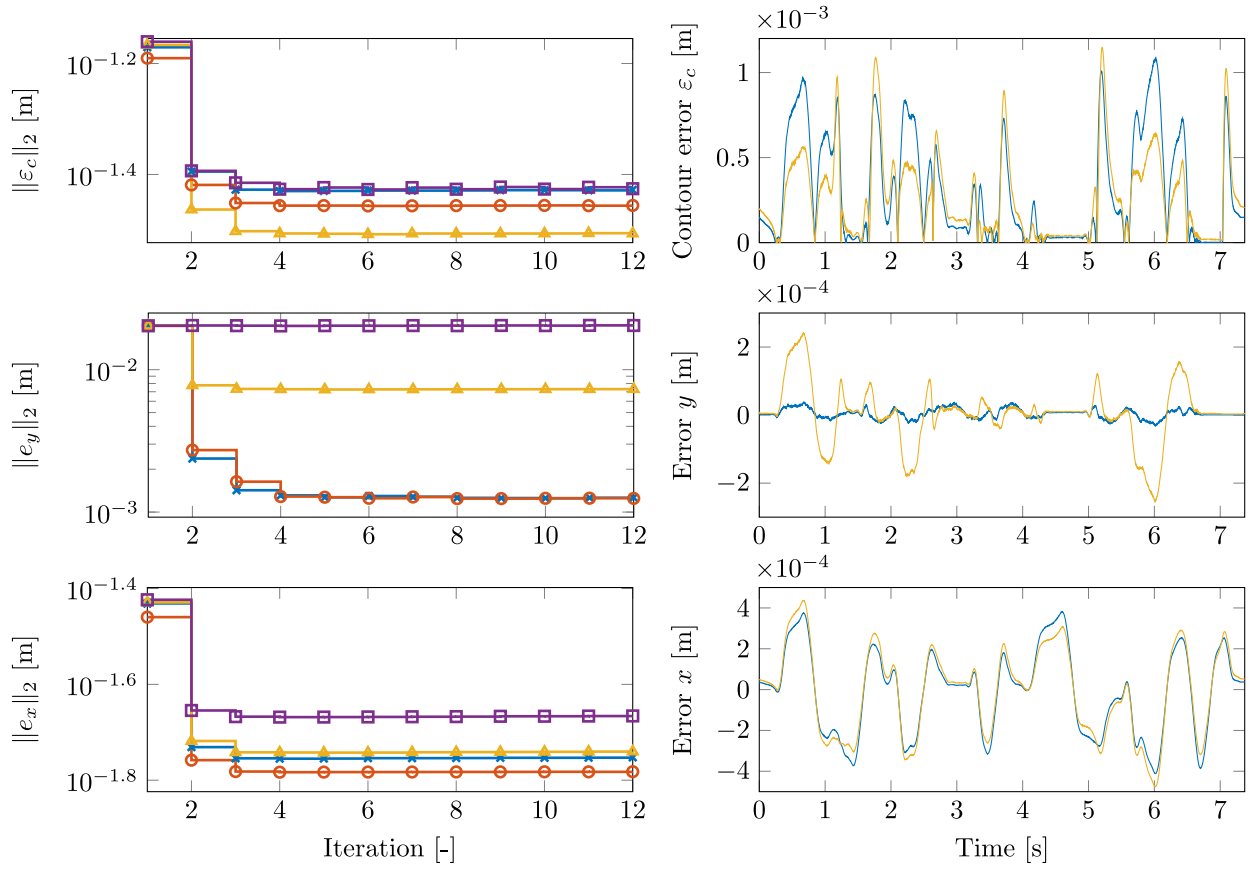
- Cross-coupled ILC with linear approximations of the contour and tangential errors:

$$C_{e,j}^k = \begin{bmatrix} \cos(\theta(k)) & -\sin(\theta(k)) \\ \sin(\theta(k)) & \cos(\theta(k)) \end{bmatrix} \forall j. \quad (36)$$

These approaches all lead to iteration-invariant weighting matrices. However, as illustrated in Fig. 2, the linear approximations are likely to differ from the exact contour error, leading to inaccuracies and reduced performance of cross-coupled ILC.

## 7. Experimental results

In this section, cross-coupled ILC is applied to an industrial flatbed printer and compared to standard norm-optimal ILC. First, the setup is introduced and second, experimental results are presented.



(a) 2-norms of the contour error (top), y-axis error (middle) and x-axis error (bottom). The weights are chosen as  $W_f = 10^{-8}I$  and  $W_{\Delta f} = 10^{-9}I$  for norm-optimal ILC with  $W_e = I$  ( $\times$ ) and cross-coupled ILC with  $W_e = I$  ( $\circ$ ),  $W_e = \text{diag}(1.5, 0.5)$  ( $\triangle$ ) and  $W_e = \text{diag}(10, 0.1)$  ( $\square$ ).

(b) Contour error (top), y-axis error (middle) and x-axis error (bottom) after 12 iterations for norm-optimal ILC with  $W_e = I$  ( $\times$ ), corresponding to  $\times$ ) and cross-coupled ILC with  $W_e = \text{diag}(1.5, 0.5)$  ( $\triangle$ ), corresponding to  $\triangle$ ).

**Fig. 7.** Experimental results for the Arizona flatbed printer. Cross-coupled ILC leads to a significant reduction of the contour error 2-norm, even if the errors of the individual axes are higher compared to standard norm-optimal ILC. Comparison of the contour error and the y-axis error show that at the time instances (e.g., between 0 and 1.5 s) where cross-coupled ILC leads to higher y-axis errors, the contour error is actually smaller compared to standard ILC.

### 7.1. Setup

The industrial flatbed printer is shown in Fig. 5 and consists of a gantry that can translate in x-direction and rotate in  $\varphi$ -direction, on top of which is a carriage that can translate in y-direction. The system is decoupled such that the input consists of the forces in three directions,  $F_x$ ,  $F_y$  and  $F_\varphi$ , and the output consists of the position in x, y and  $\varphi$ . In these experiments, the aim is to draw a contour in the y, x-plane while keeping the rotation  $\varphi$  at 0. ILC is only applied in the y- and x-directions, and since the system is fully decoupled, the system for ILC has as input  $[F_y \ F_x]^T$  and as output  $[y \ x]^T$ . The contour to be tracked is shown in Fig. 6.

### 7.2. ILC configurations

When comparing norm-optimal ILC configurations, the choice of weights is important because these strongly influence the experimental results. In this section cross-coupled ILC and standard norm-optimal ILC are compared. In both cases, parallel ILC according to Fig. 3 is used. The cost function used for standard norm-optimal ILC is referred to as  $J_{NO}$  and is given by

$$J_{NO}(f_{j+1}) = \|e_{j+1}\|_{W_e}^2 + \|f_{j+1}\|_{W_f}^2 + \|f_{j+1} - f_j\|_{W_{\Delta f}}^2, \quad (37)$$

i.e., the weights are iteration-invariant and do not include any coupling of the axes. The weighting matrices  $W_e$ ,  $W_f$  and  $W_{\Delta f}$  are time-invariant

and are given by  $W_e = w_e I$ ,  $W_f = w_f I$  and  $W_{\Delta f} = w_{\Delta f} I$ . For cross-coupled ILC, the cost function  $J_{CC}$  that is used is given by

$$J_{CC}(f_{j+1}) = \|e_{j+1}\|_{W_{ec,j}}^2 + \|f_{j+1}\|_{W_f}^2 + \|f_{j+1} - f_j\|_{W_{\Delta f}}^2, \quad (38)$$

i.e., for the weight on the error signal the outputs are coupled, but the weights on the input and change of input are decoupled. The reason for this is that weights on the input signal are typically related to actuator constraints of the system as well as to robustness against model uncertainty. For the error signal, the errors in y- and x-direction are coupled through the iteration-varying weight matrix  $W_{ec,j}$  which is of the form

$$W_{ec,j} = C_{e,j}^T W_e C_{e,j}, \quad (39)$$

with

$$W_e = \text{diag} \left( \begin{bmatrix} w_{\epsilon_c} & 0 \\ 0 & w_{\epsilon_t} \end{bmatrix} \right) \quad (40)$$

a block-diagonal, iteration- and time-invariant matrix. The iteration- and time-varying coupling matrix  $C_{e,j}$  is given by

$$C_{e,j} = \begin{bmatrix} C_{e,j}^1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & C_{e,j}^N \end{bmatrix}, \quad (41)$$



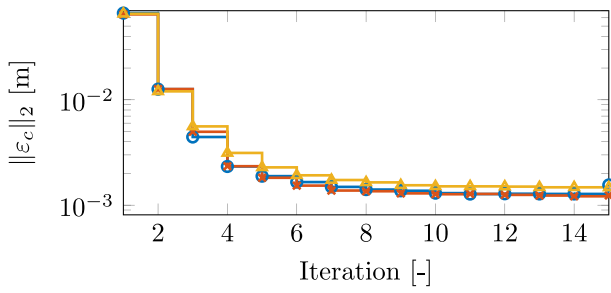


Fig. 8. 2-norm of the contour error. The weights are chosen as  $W_f = 10^{-10}I$  and  $W_{\Delta f} = 10^{-9}I$  for norm-optimal ILC with  $W_e = I$  (★) and cross-coupled ILC with  $W_e = I$  (●) and  $W_e = \text{diag}(1.5, 0.5)$  (▲). When  $W_f$  is small, the contour tracking performance of cross-coupled ILC and standard norm-optimal ILC is comparable.

with the iteration-varying matrices  $C_{e,j}^k$  given by

$$C_{e,j}^k = \begin{bmatrix} \cos(\phi(k, j)) & -\sin(\phi(k, j)) \\ \sin(\phi(k, j)) & \cos(\phi(k, j)) \end{bmatrix}, \quad (42)$$

such that

$$C_{e,j}^k \begin{bmatrix} e_y(k) \\ e_x(k) \end{bmatrix} = \begin{bmatrix} \varepsilon_c(k) \\ \varepsilon_t(k) \end{bmatrix}. \quad (43)$$

For both  $J_{NO}$  and  $J_{CC}$ ,  $W_{\Delta f} = 10^{-9}I$  is chosen. This weight on the change in input reduces the convergence speed for safety reasons, as well as the influence of iteration-varying disturbances.

### 7.3. Experimental results

In Fig. 7, standard norm-optimal ILC is compared to different cross-coupled ILC configurations. For all configurations,  $W_f = 10^{-8}I$ , and for standard norm-optimal ILC,  $W_e = I$ . For cross-coupled ILC,  $W_e = I$ ,  $W_e = \text{diag}(1.5, 0.5)$  and  $W_e = \text{diag}(10, 0.1)$  are compared, i.e., the weight is distributed differently over the contour error  $\varepsilon_c$  and the tangential error  $\varepsilon_t$ . In Fig. 7 (left), the convergence of the 2-norms of the contour error and the errors of the individual axes is compared. Fig. 7 (right) shows the converged contour and individual errors after 12 iterations. The experimental results for  $W_e = \text{diag}(1.5, 0.5)$  illustrate that cross-coupled ILC can achieve significantly smaller contour errors while the individual axis errors are much higher. The results also show that when the difference in weight between the contour error and the tangential error is increased, this does not necessarily lead to smaller contour errors. The resulting contour with and without cross-coupled ILC is shown in Fig. 6, and illustrates the overall improvement.

Next, cross-coupled ILC and standard norm-optimal ILC are compared for a case in which the weight on the input signal is smaller. In this case, the algorithms have more freedom in optimizing the input signal, and the total error can be reduced further. In Fig. 8, the 2-norm of the contour error is shown for different ILC configurations with  $W_f = 10^{-10}I$ . For standard norm-optimal ILC,  $W_e = I$  and for cross-coupled ILC,  $W_e = I$  or  $W_e = \text{diag}(1.5, 0.5)$ . The results show that when  $W_f$  is small and the input is not restricted, i.e., achieving a small contour error is feasible, cross-coupled ILC and standard norm-optimal ILC both lead to good performance and the contour errors are similar. The combination of  $W_e = \text{diag}(10, 0.1)$  with  $W_f = 10^{-10}I$  leads to feedforward inputs that violate the actuator constraints, and cannot be implemented due to safety reasons.

## 8. Conclusions

In this paper a new framework for cross-coupled norm-optimal ILC is introduced that leads to high performance by using exact contour and tangential errors in an intuitive time- and iteration-varying cost function. Conditions for the monotonic convergence of the ILC algorithm are developed. In addition, a resource-efficient implementation is

presented that interprets the ILC update law as a linear quadratic tracking problem, which can be solved fast and efficiently for any length of reference signal. The approach is validated through experiments on an industrial flatbed printer, which show that cross-coupled ILC can lead to reduced contour errors even if the errors of the individual axes are higher compared to standard norm-optimal ILC. Directions for future research include investigating the selection of iteration-varying weights in cross-coupled ILC, as well as extending the approach to the tracking of contours for systems with more than two axes.

### CRedit authorship contribution statement

**Leontine Aarnoudse:** Conceptualization, Investigation, Software, Validation, Writing – original draft, Writing – review & editing. **Johan Kon:** Conceptualization, Software, Writing – review & editing. **Koen Classens:** Conceptualization, Writing – review & editing. **Max van Meer:** Conceptualization, Writing – review & editing. **Maurice Poot:** Conceptualization, Writing – review & editing. **Paul Tacx:** Conceptualization, Writing – review & editing. **Nard Strijbosch:** Conceptualization, Writing – review & editing. **Tom Oomen:** Conceptualization, Funding acquisition, Supervision, Writing – review & editing.

### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Data availability

No data was used for the research described in the article.

### Acknowledgments

The authors gratefully acknowledge the contributions to this paper through a challenge-based learning project by Dirk Alferink, Gijs van den Brandt, Emre Deniz, Robert Devillers, Mike van Duijnhoven, Roel Habraken, Daan den Hartog, Shaun Boyteen Joseph, Jord van Kalmthout, Boudewijn Kempers, Sjoerd Leemrijse, Walter MacAulay, Paul Munns, Aron Prinsen, Stan de Rijk, Sander Ruijters, Jos Snijders, Chuck Steijlen, Jaap van der Stoel, Matthijs Teurlings, Hugo Thelosen, Peter Visser, Naomi de Vos and Matthijs van de Vosse. The authors also wish to thank Sjirk Koekebakker for his contributions.

### Appendix. Proofs of Lemma 5 and Theorem 4

First, the proof of Lemma 5 is given. Second, the proof of Theorem 4, which employs Lemma 5, is given.

**Proof of Lemma 5.** To show that (1)  $\implies$  (2), assume first that for each  $\bar{Q} \in \mathcal{Q}$  and  $\bar{L} \in \mathcal{L}$  the sequence of inputs  $\{f_j\}$  of the corresponding iteration-invariant ILC system converges monotonically to a fixed point  $\bar{f}_\infty$ , i.e.,

$$\|f_{j+1} - \bar{f}_\infty\|_2 \leq \kappa \|f_j - \bar{f}_\infty\|_2 \quad \forall j, \quad (44)$$

for some universal  $\kappa \in [0, 1)$ . The following steps are illustrated in Fig. 9.

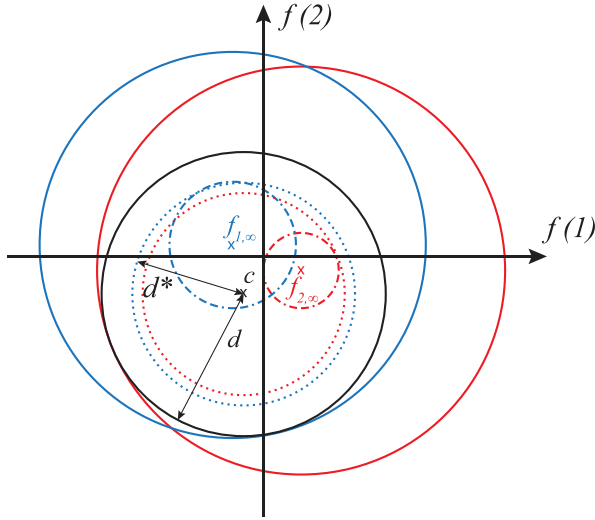
**Step 1:** Consider  $f_j \in B_2(c, d)$  and given  $Q_j \in \mathcal{Q}$ ,  $L_j \in \mathcal{L}$ .

**Step 2:** It holds that  $B_2(c, d) \subset B_2(\bar{f}_{j,\infty}, d + \|c - \bar{f}_{j,\infty}\|_2)$  because

$$\|f_j - \bar{f}_{j,\infty}\|_2 \leq \|f_j - c\|_2 + \|c - \bar{f}_{j,\infty}\|_2. \quad (45)$$

Therefore, if  $f_j \in B(c, d)$ , i.e.,  $\|f_j - c\|_2 \leq d$ , then

$$\|f_j - \bar{f}_{j,\infty}\|_2 \leq d + \|c - \bar{f}_{j,\infty}\|_2. \quad (46)$$



**Fig. 9.** Visual interpretation of steps 1–5 of the proof of Lemma 5. Step 1: the circle (—) with center  $c$  and radius  $d$ . Step 2: the circles with centers  $f_{1,\infty}$  (—) and  $f_{2,\infty}$  (—). Step 3: the circles with centers  $f_{1,\infty}$  (---) and  $f_{2,\infty}$  (---). Step 4: the circles with centers  $f_{1,\infty}$  (.....) and  $f_{2,\infty}$  (.....). Step 5: the largest circle of Step 4, i.e., the circle (.....) with center  $f_{1,\infty}$  and radius  $d^*$ .

**Step 3:** It follows from (44) that  $f_j \in B_2(\bar{f}_{j,\infty}, d + \|c - \bar{f}_{j,\infty}\|_2) \implies f_{j+1} \in B_2(\bar{f}_{j,\infty}, \kappa d + \kappa\|c - \bar{f}_{j,\infty}\|_2)$ , which leads to  $\|f_{j+1} - \bar{f}_{j,\infty}\|_2 \leq \kappa(d + \|c - \bar{f}_{j,\infty}\|_2)$ .

**Step 4:** It holds that  $B_2(\bar{f}_{j,\infty}, \kappa d + \kappa\|c - \bar{f}_{j,\infty}\|_2) \subset B_2(c, \kappa d + (1 + \kappa)\|c - \bar{f}_{j,\infty}\|_2)$  because

$$\begin{aligned} \|f_{j+1} - c\|_2 &\leq \|f_{j+1} - \bar{f}_{j,\infty}\|_2 + \|c - \bar{f}_{j,\infty}\|_2 \\ &\leq \kappa(d + \|c - \bar{f}_{j,\infty}\|_2) + \|c - \bar{f}_{j,\infty}\|_2. \end{aligned} \quad (47)$$

**Step 5:** It follows that  $f_j \in B_2(c, d) \implies f_{j+1} \in B_2(c, \kappa d + (1 + \kappa)\|c - \bar{f}_{j,\infty}\|_2) \forall Q_j \in Q, L_j \in L$ . Next, consider the set  $B(c, d^*)$  with

$$d^* = \max_{Q \in Q, L \in L} \kappa d + (1 + \kappa)\|c - \bar{f}_{j,\infty}\|_2. \quad (48)$$

Thus if  $f_j \in B_2(c, d)$ , then  $f_{j+1} \in B_2(c, d^*) \forall Q_j \in Q, L_j \in L$ .

**Step 6:** Since the sets  $Q$  and  $L$  are finite and  $\kappa \in [0, 1)$ , there exists  $a$  such that  $d > d^*$  if  $d > a$ , and  $d = d^*$  if  $d = a$ .

To show that (2)  $\implies$  (1), assume that the sequence of inputs  $\{f_j\}$  is monotonically convergent in the 2-norm to a closed 2-norm ball given by  $B_2(c, a)$ , i.e.,

$$\|f_{j+1} - c\|_2 \leq \kappa\|f_j - c\|_2 \text{ if } \|e_j - c\|_2 > a, \quad (49)$$

Since this is satisfied for any iteration-varying  $Q_j \in Q, L_j \in L$ , the iteration-invariant case where  $Q_j = \bar{Q}, L_j = \bar{L} \forall j$  also satisfies (49). Therefore, each of the iteration-invariant systems converges monotonically towards the closed 2-norm ball  $B_2(c, a)$  and since they are iteration-invariant, they converge monotonically to a fixed point in this set.  $\square$

**Proof of Theorem 4.** The proof consists of four steps.

**Step 1.** The sequence of inputs  $\{f_j\}$  in (19) for fixed  $\bar{Q} \in Q$  and  $\bar{L} \in L$  is monotonically convergent in the 2-norm if the mapping from  $f_j$  to  $f_{j+1}$  is a contraction mapping according to the Banach fixed point

theorem [22, Theorem 5.1-2]. Substituting (1) in (19) shows that this is satisfied if

$$\|\bar{Q} - \bar{L}J\|_2 < 1. \quad (50)$$

**Step 2.** It holds that  $\bar{\sigma}(\bar{Q} - \bar{L}J) \leq \|\bar{Q} - \bar{L}J\|_2$ . From (21) it follows that (50) is satisfied if

$$\bar{\sigma}(\bar{Q} - \bar{L}J) = \bar{\sigma}((J^\top \bar{W}_{ec} J + \bar{W}_{fc} + \bar{W}_{dfc})^{-1} \bar{W}_{dfc}) < 1.$$

It holds that  $\bar{\sigma}((A + B)^{-1}B) < 1$  for  $A > 0, B \geq 0$ . For a positive (semi)definite matrix  $M$ ,  $A^\top M A$  is positive (semi)definite if  $A$  has full column rank. Thus for non-singular  $J$ ,  $\bar{W}_{ec} > 0, \bar{W}_{fc}, \bar{W}_{dfc} \geq 0$  ensures monotonic convergence. For singular  $J$ ,  $\bar{W}_{fc} > 0$  is needed also.

**Step 3.** Matrices  $W_{ec}, W_{fc}$  and  $W_{dfc}$  are structured as  $C^\top W C$ . Thus,  $\bar{W}_{ec}, \bar{W}_{fc} > 0$  is satisfied if  $C_e$  respectively  $C_f$  has full column rank and  $W_e$  respectively  $W_f > 0$ . Additionally,  $\bar{W}_{dfc} \geq 0$  is satisfied for  $W_{dfc} \geq 0$ .

**Step 4.** Applying Step 1–3 for each  $Q_j \in Q, L_j \in L$  and combining with Lemma 5 concludes the proof.  $\square$

It is also possible to find an expression for the smallest closed 2-norm ball to which the system converges, a result used in a preliminary version of [23], see [24, Theorem III.9].

## References

- [1] Jia Z-y, Ma J-w, Song D-n, Wang F-j, Liu W. A review of contouring-error reduction method in multi-axis CNC machining. *Int J Mach Tools Manuf* 2018;125:34–54. <http://dx.doi.org/10.1016/j.ijmachtools.2017.10.008>.
- [2] Yang S, Ghasemi AH, Lu X, Okwudire CE. Pre-compensation of servo contour errors using a model predictive control framework. *Int J Mach Tools Manuf* 2015;98:50–60. <http://dx.doi.org/10.1016/j.ijmachtools.2015.08.002>.
- [3] Koren Y. Cross-coupled biaxial computer control for manufacturing systems. *J Dyn Syst Meas Control* 1980;102:265–72. <http://dx.doi.org/10.1115/1.3149612>.
- [4] Shih YT, Chen CS, Lee AC. A novel cross-coupling control design for bi-axis motion. *Int J Mach Tools Manuf* 2002;42(14):1539–48. [http://dx.doi.org/10.1016/S0890-6955\(02\)00109-8](http://dx.doi.org/10.1016/S0890-6955(02)00109-8).
- [5] Huo F, Poo AN. Improving contouring accuracy by using generalized cross-coupled control. *Int J Mach Tools Manuf* 2012;63:49–57. <http://dx.doi.org/10.1016/j.ijmachtools.2012.07.012>.
- [6] Gunnarsson S, Norrlöf M. On the design of ILC algorithms using optimization. *Automatica* 2001;37(12):2011–6. [http://dx.doi.org/10.1016/S0005-1098\(01\)00154-6](http://dx.doi.org/10.1016/S0005-1098(01)00154-6).
- [7] Owens DH. Iterative learning control: An optimization paradigm. London: Springer-Verlag; 2016. <http://dx.doi.org/10.1007/978-1-4471-6772-3>.
- [8] Chen HR, Cheng MY, Wu CH, Su KH. Real time parameter based contour error estimation algorithms for free form contour following. *Int J Mach Tools Manuf* 2016;102. <http://dx.doi.org/10.1016/j.ijmachtools.2015.11.009>.
- [9] van Zundert J, Bolder J, Koekebakker S, Oomen T. Resource-efficient ILC for LTI/LTV systems through LQ tracking and stable inversion: Enabling large feedforward tasks on a position-dependent printer. *Mechatronics* 2016;38:76–90. <http://dx.doi.org/10.1016/j.mechatronics.2016.07.001>.
- [10] Barton K, Alleyne A. A cross-coupled iterative learning control design for precision motion control. *IEEE Trans Control Syst Technol* 2008;16(6):1218–31. <http://dx.doi.org/10.1109/TCST.2008.919433>.
- [11] Barton KL, Alleyne AG. A norm optimal approach to time-varying ILC with application to a multi-axis robotic testbed. *IEEE Trans Control Syst Technol* 2011;19(1):166–80. <http://dx.doi.org/10.1109/TCST.2010.2040476>.
- [12] Sun H, Alleyne AG. A cross-coupled non-lifted norm optimal iterative learning control approach with application to a multi-axis robotic testbed. In: 19th IFAC world congr.. 2014, p. 2046–51. <http://dx.doi.org/10.3182/20140824-6-za-1003.00519>.
- [13] Sun H, Alleyne AG. A computationally efficient norm optimal iterative learning control approach for LTV systems. *Automatica* 2014;50(1):141–8. <http://dx.doi.org/10.1016/j.automat.2014.08.019>.
- [14] Chen SL, Hsieh SM. Iterative learning contouring control: Theory and application to biaxial systems. *Mechatronics* 2023;89:102932. <http://dx.doi.org/10.1016/j.mechatronics.2022.102932>.
- [15] Aarnoudse L, Kon J, Classens K, van Meer M, Poot M, Tacx P, Strijbosch N, Oomen T. Cross-coupled iterative learning control for complex systems: A monotonically convergent and computationally efficient approach. In: Conf. decis. control. Cancún, Mexico; 2022. <http://dx.doi.org/10.1109/CDC51059.2022.9993408>.

- [16] Amann N, Owens DH, Rogers E. Iterative learning control for discrete-time systems with exponential rate of convergence. *IEEE Proc Control Theory Appl* 1996;143(2):217–24. <http://dx.doi.org/10.1049/ip-cta:19960244>.
- [17] Oomen T, Rojas CR. Sparse iterative learning control with application to a wafer stage: Achieving performance, resource efficiency, and task flexibility. *Mechatronics* 2017;47:134–47. <http://dx.doi.org/10.1016/j.mechatronics.2017.09.004>.
- [18] Lewis FL, Vrabie D, Vamvoudakis KG. Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers. *IEEE Control Syst Mag* 2012;32(6):76–105. <http://dx.doi.org/10.1109/MCS.2012.2214134>.
- [19] Ebrahimpzadeh F, Tsai JSH, Chung MC, Liao YT, Guo SM, Shieh LS, Wang L. A generalised optimal linear quadratic tracker with universal applications. Part 2: discrete-time systems. *Internat J Systems Sci* 2017;48(2):397–416. <http://dx.doi.org/10.1080/00207721.2016.1186240>.
- [20] Oomen T, van de Wijdeven J, Bosgra OH. System identification and low-order optimal control of intersample behavior in ILC. *IEEE Trans Automat Control* 2011;56(11):2734–9. <http://dx.doi.org/10.1109/TAC.2011.2160596>.
- [21] Barton KL, Hoelzle DJ, Alleyne AG, Johnson AJ. Cross-coupled iterative learning control of systems with dissimilar dynamics: Design and implementation. *Internat J Control* 2011;84(7):1223–33. <http://dx.doi.org/10.1080/00207179.2010.500334>.
- [22] Kreyszig E. *Introductory functional analysis with applications*. John Wiley & Sons, Inc.; 1978.
- [23] Strijbosch N, Oomen T. Iterative learning control for intermittently sampled data: Monotonic convergence, design, and applications. *Automatica* 2022;139:110171. <http://dx.doi.org/10.1016/j.automatica.2022.110171>.
- [24] Strijbosch N, Oomen T. Beyond quantization in iterative learning control: Exploiting time-varying time-stamps. In: *Proc. am. control conf.*. Philadelphia, PA, USA; 2019, p. 2984–9. <http://dx.doi.org/10.23919/acc.2019.8815329>.



**Leontine Aarnoudse** received the B.Sc. degree (2017) and M.Sc. degree (cum laude) (2019) in Mechanical Engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands. She is currently pursuing a Ph.D. degree in the Control Systems Technology group within the department of Mechanical Engineering at Eindhoven University of Technology. Her research interests are in the field of control for precision mechatronics, and are mostly centered around the development of learning theory for these systems.



**Johan Kon** received the B.Sc. degree (cum laude, 2018) in Mechanical Engineering and M.Sc. degree (cum laude, 2021) in Systems & Control from the Eindhoven University of Technology. He is currently pursuing a Ph.D. degree in the Control Systems Technology group within the department of Mechanical Engineering at Eindhoven University of Technology. His research interests include machine learning for system identification and control applied to precision mechatronics.



**Koen Classens** received the B.Sc. degree (cum laude) (2016), the M.Sc. degree (cum laude) (2019) in Mechanical Engineering, and the M.Sc. degree (cum laude) (2019) in Systems & Control from the Eindhoven University of Technology, Eindhoven, The Netherlands. He is currently pursuing a Ph.D. degree in the Control Systems Technology group within the department of Mechanical Engineering at Eindhoven University of Technology. His research is centered around the development of fault detection and isolation systems for precision mechatronics.



**Max van Meer** received the M.Sc. degree (cum laude) in Mechanical Engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands in 2021. He is currently a Ph.D. candidate at the Eindhoven University of Technology in the Department of Mechanical Engineering. His research interests include learning control, motor commutation and machine learning for motion control.



**Maurice Poot** received the M.Sc. degree from the Eindhoven University of Technology, Eindhoven, The Netherlands in 2019. He is currently a Ph.D. candidate at the Eindhoven University of Technology in the Department of Mechanical Engineering. His research interests include Gaussian processes, data-driven learning, learning, and control with application to mechatronic systems.



**Paul Tacx** received the M.Sc. degree (cum laude) in mechanical engineering from the Eindhoven University of Technology, Eindhoven, the Netherlands, in 2019, where he is currently pursuing the Ph.D. degree with the Control Systems Technology Group. His research interests include identification for advanced motion control and control of complex mechatronic systems.



**Nard Strijbosch** received his B.Sc. degree (cum laude), M.Sc. degree (cum laude) and Ph.D. degree from the Eindhoven University of Technology, Eindhoven, The Netherlands. He is currently a system engineer at IBS Precision Engineering. He is a recipient of the IEEE Industry Applications Society Excellent Presentation Award (SAMCON 2018). His research interest is in the field of motion control and learning control techniques for applications in mechatronic systems.



**Tom Oomen** is full professor with the Department of Mechanical Engineering at the Eindhoven University of Technology. He is also a part-time full professor with the Delft University of Technology. He received the M.Sc. degree (cum laude) and Ph.D. degree from the Eindhoven University of Technology, Eindhoven, The Netherlands. He held visiting positions at KTH, Stockholm, Sweden, and at The University of Newcastle, Australia. He is a recipient of the 7th Grand Nagamori Award, the Corus Young Talent Graduation Award, the IFAC 2019 TC 4.2 Mechatronics Young Research Award, the 2015 IEEE Transactions on Control Systems Technology Outstanding Paper Award, the 2017 IFAC Mechatronics Best Paper Award, the 2019 IEEE Journal of Industry Applications Best Paper Award, and recipient of a Veni and Vidi personal grant. He is currently a Senior Editor of IEEE Control Systems Letters (L-CSS) and Associate Editor of IFAC Mechatronics, and he has served on the editorial boards of the IEEE Control Systems Letters (L-CSS) and IEEE Transactions on Control Systems Technology. He has also been vice-chair for IFAC TC 4.2 and a member of the Eindhoven Young Academy of Engineering. His research interests are in the field of data-driven modeling, learning, and control, with applications in precision mechatronics.