

# Cross-Coupled Iterative Learning Control for Complex Systems: A Monotonically Convergent and Computationally Efficient Approach\*

Leontine Aarnoudse<sup>1</sup>, Johan Kon<sup>1</sup>, Koen Classens<sup>1</sup>, Max van Meer<sup>1</sup>,  
Maurice Poot<sup>1</sup>, Paul Tacx<sup>1</sup>, Nard Strijbosch<sup>2</sup> and Tom Oomen<sup>1,3</sup>

**Abstract**—Cross-coupled iterative learning control (ILC) can achieve high performance for manufacturing applications in which tracking a contour is essential for the quality of a product. The aim of this paper is to develop a framework for norm-optimal cross-coupled ILC that enables the use of exact contour errors that are calculated offline, and iteration- and time-varying weights. Conditions for the monotonic convergence of this iteration-varying ILC algorithm are developed. In addition, a resource-efficient implementation is proposed in which the ILC update law is reframed as a linear quadratic tracking problem, reducing the computational load significantly. The approach is illustrated on a simulation example.

## I. INTRODUCTION

The demands for accuracy and speed in manufacturing are ever increasing and necessitate improved control strategies. In many multiple-input multiple-output (MIMO) applications such as (3D) printing and CNC machining, accurately following a contour is essential for the quality of the final product. For increasingly complex parts with curved surfaces and sharp curvature variation, high feedrates lead to high contour errors, resulting in low processing qualities and efficiency [1]. Standard control approaches that aim at following a time-based reference are not suitable for this type of applications.

Approaches to minimize contour errors include offline methods such as trajectory generation and pre-compensation [2] and online methods such as improved control of individual axes and cross-coupled feedback control [3], [4]. Cross-coupled feedback control uses an online estimate of the contour error, which is typically a reference-dependent combination of the individual axes errors. This introduces time-varying couplings between the system axes, leading to linear time-varying (LTV) or non-linear systems, for which feedback control is not straightforward [5]. In addition, stability and computation times limit the accuracy of online contour error estimates for cross-coupled feedback control.

Iterative learning control (ILC) is capable of achieving high accuracy by updating a feedforward signal iteratively based on repeated experiments [6]. Updating the input using previous error signals leads to high performance after only a small number of experiments. Typical frameworks include

frequency-domain [7] and lifted norm-optimal ILC [8], [9]. Typical norm-optimal ILC aims at reducing the deviation from a time-based reference signal for individual axes by minimizing a cost function that includes individual axes error and input signals. While this leads to high performance for the individual axes, standard ILC has limitations for contour tracking applications. In particular, it 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. For contour tracking, the performance of ILC can be increased significantly by explicitly taking the contour error into account.

Several aspects are of importance for cross-coupled ILC. First, high accuracy of the contour error estimate is a requirement for precise control. Online contour error estimates vary in accuracy and computational load, and range from linear or circular [3] to more complex parameter-based approximations [10]. Since feedforward signals in ILC are calculated offline, the exact contour error can be used instead of these approximations. The exact contour error is constructed from the individual axes errors through coupling gains that follow from comparing an output coordinate to each point on the reference contour. These exact coupling gains are not only time-varying but also iteration-varying, and as such require an ILC framework that allows for variation over iterations.

Second, cross-coupled control should consider not only the contour error but also a tangential error to ensure that the system moves over the contour. For norm-optimal cross-coupled ILC, it is essential that the tuning of the design parameters of the algorithm is intuitive and reflects the trade-off between accuracy and speed represented by respectively the contour and tangential errors. In addition, the relative importance of contour and tangential errors typically differs over the reference, e.g., in corners accuracy is more important than speed. The framework should enable time-varying weighting to allow for these variations. An approach to cross-coupled ILC is proposed in [11], in which standard ILC and feedback control for the individual axes are combined with PD-based ILC based on the contour error. In [12] the method is extended to norm-optimal ILC with time-varying weighting. Both approaches use linear contour error estimates, limiting the achievable performance. In addition, while the norm-optimal ILC approach weights both individual axes and contour errors, the tuning is not intuitive since the trade-off between tangential and contour errors is not made explicit.

Third, the considered MIMO applications often use long reference signals. However, standard norm-optimal lifted ILC

\*This work is part of the research programme VIDI with project number 15698, which is (partly) financed by the NWO.

<sup>1</sup>The authors are with the Dept. of Mechanical Engineering, Control Systems Technology, Eindhoven University of Technology, Eindhoven, The Netherlands. l.i.m.aarnoudse@tue.nl

<sup>2</sup>Nard Strijbosch is with IBS Precision Engineering, Eindhoven, The Netherlands

<sup>3</sup>Tom Oomen is also with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands.

methods are limited to reference signals with a relatively small number of samples, because the ILC update law involves matrix inversions for which the computational load scales badly with the signal length [13]. For cross-coupled ILC, a low-order solution with efficient computations that allows for long reference signals is essential. The implementation in [12] requires inversion of large matrices and the approach is therefore limited to short reference signals. To remove restrictions on the reference length, non-lifted cross-coupled ILC is developed in [14]. This approach uses linear contour error approximations in individual cost functions at each error sample, and results in a PD-like ILC controller that does not allow for a direct feedthrough term in the plant.

Although several important steps have been taken towards cross-coupled ILC for contour tracking, a framework that uses exact contour errors and enables intuitive tuning and unlimited signal lengths is lacking. This paper aims to address these aspects, resulting in the following contributions.

- A cost function is introduced that allows the use of exact contour errors with iteration-varying coupling matrices, and that enables intuitive tuning of time- and iteration-varying weights (Section III).
- Conditions for monotonic convergence of the ILC algorithm with the proposed iteration-varying cost function are given (Section IV).
- A resource-efficient implementation based on linear quadratic tracking is proposed that allows for inexpensive and fast computations in case of iteration-varying ILC matrices and long reference signals (Section V).

The approach is illustrated using a simulated flatbed printer in Section VI. Conclusions are given in Section VII.

## II. PROBLEM FORMULATION

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

### A. Contour tracking

Consider a discrete-time, linear time-varying (LTV) MIMO system with  $n_i$  inputs and  $n_o$  outputs. For reference  $y_d$ , the tracking error  $e$  is given by

$$e = Sy_d - Jf, \quad (1)$$

with sensitivity  $S = (I + PC)^{-1}$  for plant  $P$  and controller  $C$ , process sensitivity  $J = P(I + CP)^{-1}$ , and feedforward input  $f$  as shown in Fig. 1. 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 error  $e \in \mathbb{R}^{Nn_o \times 1}$  is written similar to input  $f \in \mathbb{R}^{Nn_i \times 1}$ , and  $e(k) \in \mathbb{R}^{n_o \times 1}$  and  $f(k) \in \mathbb{R}^{n_i \times 1}$  are of the form  $e(k) =$

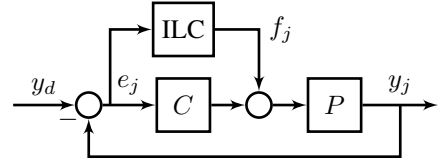


Fig. 1: Parallel ILC configuration.

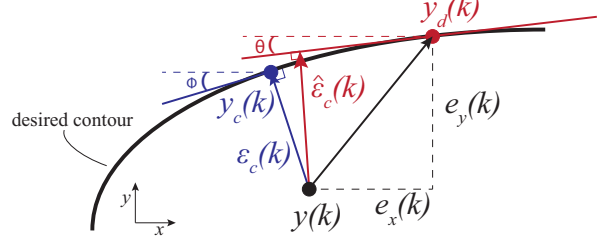


Fig. 2: For a 2D-system with reference  $y_d(k)$ , position output  $y(k)$  and individual axes errors  $e_x(k)$  and  $e_y(k)$ , the actual contour error  $\varepsilon_c(k)$  (—) based on the closest point  $y_c(k)$  on the contour differs from linear approximation  $\hat{\varepsilon}_c(k)$  (—).

$[e^1(k) \ e^2(k) \ \dots \ e^{n_o}(k)]^T$ . The aim of the system is to track a contour described by the reference  $y_d(k) \in \mathbb{R}^{n_o \times 1}$  accurately in space rather than in time. To that end, in addition to the time-based error  $e(k)$ , the contour error  $\varepsilon_c(k) \in \mathbb{R}$  is defined as the distance between the position output  $y(k) \in \mathbb{R}^{n_o \times 1}$  and the closest point on the contour, as illustrated in Fig. 2. Each contour error sample  $\varepsilon_c(k)$  can be expressed as a function of  $e(k)$  through a vector of coupling gains  $c(k) \in \mathbb{R}^{n_o \times 1}$  according to

$$\varepsilon_c(k) = c(k)^T e(k). \quad (2)$$

The coupling gains depend on the type of approximation used to determine  $\varepsilon_c$ , as is further explained in Section III.

### B. Norm-optimal iterative learning control

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

$$e_j = Sy_d - Jf_j, \quad (3)$$

$$f_{j+1} = Qf_j + Le_j, \quad (4)$$

where error  $e_j$  and  $f_j$  depend on iteration  $j$ . The update matrices  $Q$  and  $L$  follow from the cost function

$$\mathcal{J}(f_{j+1}) = \|e_{j+1}\|_{W_e}^2 + \|f_{j+1}\|_{W_f}^2 + \|f_{j+1} - f_j\|_{W_{\Delta f}}^2, \quad (5)$$

with  $\|x\|_W = \sqrt{x^T W x}$ . The minimizer of  $\mathcal{J}(f_{j+1})$  can be determined analytically [8] and leads to

$$Q = (J^T W_e J + W_f + W_{\Delta f})^{-1} (J^T W_e J + W_{\Delta f}) \quad (6)$$

$$L = (J^T W_e J + W_f + W_{\Delta f})^{-1} J^T W_e. \quad (7)$$

If the convolution matrix  $J$  is non-singular, norm-optimal ILC leads to monotonic convergence of the sequence of input signals  $\{f_j\}$  for  $W_e \succ 0$ ,  $W_f, W_{\Delta f} \succeq 0$ . If  $J$  is singular then  $W_f \succ 0$  enforces monotonic convergence.

This paper aims to develop a framework for norm-optimal cross-coupled ILC that achieves high performance for contour tracking applications. A time- and iteration-varying cost

function is developed that allows the use of exact contour errors as well as various estimations and that enables intuitive tuning. The convergence is analyzed and a resource-efficient implementation suitable for long references is proposed.

### III. COST FUNCTION DESIGN

In this section, a cost function is introduced that encompasses different configurations of cross-coupled ILC. It enables using exact contour errors with iteration-varying coupling gains and intuitive time- and iteration-varying weights.

#### A. Cost function framework

The following cost function is proposed, in which different approaches to cross-coupled ILC can be incorporated

$$\mathcal{J}(f_{j+1}) = \|e_{j+1}\|_{W_{e,c,j}}^2 + \|f_{j+1}\|_{W_{f,c,j}}^2 + \|f_{j+1} - f_j\|_{W_{\Delta f,c,j}}^2, \quad (8)$$

The weighting-coupling matrices  $W_{e,c,j}$ ,  $W_{f,c,j}$  and  $W_{\Delta f,c,j}$  may be iteration-varying and can take into account individual axes inputs and errors, different approximations of the contour error, and the tangential error that is perpendicular to the contour error. The matrices are of the form

$$W_{e,c,j} = C_{e,j}^T W_{e,j} C_{e,j}, \quad W_{f,c,j} = C_{f,j}^T W_{f,j} C_{f,j}, \quad (9)$$

$$W_{\Delta f,c,j} = C_{f,j}^T W_{\Delta f,j} C_{f,j}$$

where the coupling matrices  $C_{e,j}$  for  $e$  and  $C_{f,j}$  for  $f$  may differ, for example when the MIMO system is non-square or different couplings for input and error are desired. The block-diagonal matrices are constructed as

$$C_{e,j} = \begin{bmatrix} C_{e,j}^1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & C_{e,j}^N \end{bmatrix}. \quad (10)$$

The blocks  $C_{e,j}^k \in \mathbb{R}^{n_{ce} \times n_o}$ , with  $n_{ce}$  the number of coupled error components, describe the coupling of the axes at each sample and have full column rank to ensure that each output component is taken into account. Standard norm-optimal ILC is recovered for  $C_{e,j}^k = I^{n_i \times n_o} \forall k, j$ . Possible configurations for a 2D system with  $e(k) = [e_y(k) \quad e_x(k)]^T$  include:

- Cross-coupled ILC with linear approximations of the contour error  $\hat{\varepsilon}_c$  and individual axes errors, with  $\theta(k)$  the angle between reference sample  $y_d(k)$  and the  $x$ -axis:

$$C_{e,j}^k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \cos(\theta(k)) & -\sin(\theta(k)) \end{bmatrix} \forall j, \quad (11)$$

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

$$C_{e,j}^k = \begin{bmatrix} \cos(\theta(k)) & -\sin(\theta(k)) \\ \sin(\theta(k)) & \cos(\theta(k)) \end{bmatrix} \forall j. \quad (12)$$

- Cross-coupled ILC with exact contour and tangential errors, with  $\phi(k, j)$  the angle between the  $x$ -axis and the vector perpendicular to the contour error vector. The contour error vector is the vector from position  $y_j(k)$  to the closest point on the contour  $y_{c,j}(k)$ , see Fig. 2. In this

approach, the coupling gains may differ over iterations, resulting in iteration-varying weighting-coupling matrices:

$$C_{e,j}^k = \begin{bmatrix} \cos(\phi(k, j)) & -\sin(\phi(k, j)) \\ \sin(\phi(k, j)) & \cos(\phi(k, j)) \end{bmatrix}, \quad (13)$$

The matrix  $C_{f,j}$  with blocks  $C_{f,j}^k \in \mathbb{R}^{n_{cf} \times n_o}$  is designed similar to  $C_{e,j}$ . For systems with  $n_o > 2$  the exact contour error definition can be extended easily, but approximations become more complicated [1].

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

The structure of cost function (8) allows for time-varying weighting matrices  $W_{e,j}$ ,  $W_{f,j}$  and  $W_{\Delta f,j}$ . The matrices have the following block-diagonal structure:

$$W_{e,j} = \begin{bmatrix} W_{e,j}^1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & W_{e,j}^N \end{bmatrix}, \quad (14)$$

where the size of  $W_{e,j}^k \in \mathbb{R}^{n_{ce} \times n_{ce}}$  depends on that of the corresponding block  $C_{e,j}^k$  of the coupling matrix. The diagonal blocks  $W_{e,j}^k$  apply weights to each element of the coupled error. For example, for  $C_{e,j}^k$  according to (13), i.e., exact contour and tangential errors, it holds that

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

such that the first diagonal element 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 blocks  $W_{e,j}^k$  need not be identical for all  $k$ , allowing time-varying weighting such as increased weights on the contour error in corners to increase accuracy. The weights may also be iteration-varying, e.g., one could weight only individual axes errors initially and add weights on the contour error after some iterations.

### IV. MONOTONIC CONVERGENCE

In this section, conditions are developed for the monotonic convergence towards a closed 2-norm ball of cross-coupled ILC with the iteration-varying cost function of Section III. The ILC update based on cost function (8) is given by

$$f_{j+1} = Q_j f_j + L_j e_j, \quad \text{with} \quad (16)$$

$$Q_j = (J^T W_{e,c,j} J + W_{e,f,j} + W_{\Delta f,c,j})^{-1} (J^T W_{e,c,j} J + W_{\Delta f,c,j}), \quad (17)$$

$$L_j = (J^T W_{e,c,j} J + W_{e,f,j} + W_{\Delta f,c,j})^{-1} J^T W_{e,c,j}. \quad (18)$$

The sets containing all possible filters  $Q_j$  and  $L_j$  are given by  $\mathcal{Q}$  and  $\mathcal{L}$ , respectively. Since the reference and position are sampled with finite resolution, the sets  $\mathcal{Q}$  and  $\mathcal{L}$  are finite provided that the set of iteration-varying weights is also finite. First, monotonic convergence towards a closed 2-norm ball is defined, then a convergence theorem is given.

**Definition 1** (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 2** (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  $i \in \mathbb{Z}_{\geq 0}$ ,*

$$\|y_{i+1} - c\|_2 \leq \kappa \|y_i - c\|_2 \quad \text{if } y_i \notin B_2(c, d), \quad (19)$$

$$y_{j+1} \in B_2(c, d) \quad \text{if } y_i \in B_2(c, d). \quad (20)$$

**Theorem 3.** *The sequence of inputs  $\{f_j\}$  that follows from update law (16), with iteration-varying filters  $Q_j \in \mathcal{Q}$  and  $L_j \in \mathcal{L}$  that minimize criterion (8) according to (18), is monotonically convergent towards a closed 2-norm ball if the coupling matrix  $C_{e,j}$  has full column rank,  $W_{e,j} \succ 0 \forall j$  and either,*

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

The following auxiliary lemma is used in the proof.

**Lemma 4.** *For iteration (16), 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.*

*Proof of Lemma 4.* 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.,

$$\|f_{j+1} - \bar{f}_\infty\|_2 \leq \kappa \|f_j - \bar{f}_\infty\|_2 \quad \forall j, \quad (21)$$

for some universal  $\kappa \in [0, 1)$ . Consider  $f_j \in B_2(c, d)$  and given  $Q_j \in \mathcal{Q}, L_j \in \mathcal{L}$ . It holds that  $B_2(c, d) \subset B_2(\bar{f}_{j,\infty}, d + \|c - \bar{f}_{j,\infty}\|_2)$  because

$$\|f_j - \bar{f}_{j,\infty}\|_2 \leq \|f_j - c\|_2 + \|c - \bar{f}_{j,\infty}\|_2. \quad (22)$$

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

$$\|f_j - \bar{f}_{j,\infty}\|_2 \leq d + \|c - \bar{f}_{j,\infty}\|_2. \quad (23)$$

It follows from (21) 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

$$\|f_{j+1} - \bar{f}_{j,\infty}\|_2 \leq \kappa(d + \|c - \bar{f}_{j,\infty}\|_2).$$

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

$$\begin{aligned} \|f_{j+1} - c\|_2 &\leq \|f_{j+1} - \bar{f}_{j,\infty}\|_2 + \|c - \bar{f}_{j,\infty}\|_2 \\ &\leq \kappa(d + \|c - \bar{f}_{j,\infty}\|_2) + \|c - \bar{f}_{j,\infty}\|_2. \end{aligned} \quad (24)$$

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

$$d^* = \max_{Q \in \mathcal{Q}, L \in \mathcal{L}} \kappa d + (1 + \kappa)\|c - \bar{f}_\infty\|_2. \quad (25)$$

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

$$\|f_{j+1} - c\|_2 \leq \kappa \|f_j - c\|_2 \quad \text{if } \|e_j - c\|_2 > a, \quad (26)$$

Since this is satisfied for any iteration-varying  $Q_j \in \mathcal{Q}, L_j \in \mathcal{L}$ , the iteration-invariant case where  $Q_j = \bar{Q}, L_j = \bar{L} \forall j$  also satisfies (26). 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 3.* The proof consists of four steps.

*Step 1.* The sequence of inputs  $\{f_j\}$  in (16) for fixed  $\bar{Q} \in \mathcal{Q}$  and  $\bar{L} \in \mathcal{L}$  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 [15, Theorem 5.1-2]. Substituting (1) in (16) shows that this is satisfied if

$$\|\bar{Q} - \bar{L}J\|_2 < 1. \quad (27)$$

*Step 2.* It holds that  $\bar{\sigma}(\bar{Q} - \bar{L}J) \leq \|\bar{Q} - \bar{L}J\|_2$ . From (18) it follows that (27) is satisfied if

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

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

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

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

Note that for iteration-invariant weighting-coupling matrices, Theorem 3 ensures monotonic convergence in the 2-norm to a fixed point instead. Theorem 3 reduces the design of cross-coupled ILC for monotonic convergence to choosing suitable weights and couplings, enabling intuitive design. It is 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 [16], see [17, Theorem III.9].

## V. RESOURCE-EFFICIENT IMPLEMENTATION

In this section, the cross-coupled ILC update law (16) is reframed as a linear quadratic tracking problem with a resource-efficient solution, reducing the computational load significantly and enabling long reference signals. This is especially useful for iteration-varying cost functions, since  $Q_j$  and  $L_j$  are not calculated explicitly and the inversion in (18), that would otherwise limit the size of the lifted matrices and thus the length of the reference signal, is avoided.

Process sensitivity  $J$  is rewritten to state-space description

$$\mathbf{J}_{SP} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}. \quad (28)$$

Due to the block-diagonal structure of  $W_{ec,j}$ ,  $W_{fc,j}$  and  $W_{\Delta fc,j}$  the cost function (8) can be written as

$$\begin{aligned} \mathcal{J}(f_{j+1}) = & \sum_{k=1}^N \|e_{j+1}(k)\|_{W_{ec,j}^k}^2 + \|f_{j+1}(k)\|_{W_{fc,j}^N}^2 \\ & + \|f_{j+1}(k) - f_j(k)\|_{W_{\Delta fc,j}^k}^2, \end{aligned} \quad (29)$$

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 (3) 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 (8) to the solution of a linear quadratic tracking problem.

**Theorem 5.** *The optimal ILC input that minimizes (8) with  $W_{\Delta fc,j} \succ 0$  is the solution to the linear quadratic tracking problem with cost function*

$$\mathcal{J}(\Delta f_{j+1}) = \sum_{k=1}^N \Delta f_{j+1}^T(k) \underbrace{W_{\Delta fc,j}^k}_{R_j^k} \Delta f_{j+1}(k) + \quad (30)$$

$$\begin{bmatrix} e_j(k) + \Delta e_{j+1}(k) \\ f_j(k) + \Delta f_{j+1}(k) \end{bmatrix}^T \underbrace{\begin{bmatrix} W_{ec,j}^k & 0 \\ 0 & W_{fc,j}^k \end{bmatrix}}_{S_j^k} \begin{bmatrix} e_j(k) + \Delta e_{j+1}(k) \\ f_j(k) + \Delta f_{j+1}(k) \end{bmatrix},$$

subject to the dynamics

$$\Delta x_{j+1}(k+1) = A \Delta x_{j+1}(k) + B \Delta f_{j+1}(k), \quad (31)$$

$$\begin{aligned} \Delta y_{j+1}(k) &= \begin{bmatrix} \Delta e_{j+1}(k) \\ \Delta f_{j+1}(k) \end{bmatrix} \\ &= \begin{bmatrix} C \\ 0 \end{bmatrix} \Delta x_{j+1}(k) + \begin{bmatrix} D \\ I \end{bmatrix} \Delta f_{j+1}(k). \end{aligned} \quad (32)$$

*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 (29). The problem is reframed as an LQT problem with direct feedthrough by adding output  $\Delta f_{j+1}$ . Taking  $W_{\Delta fc,j} \succ 0$  ensures that  $S_j^k \succeq 0$ ,  $R_j^k \succ 0$  and concludes the proof.  $\square$

The solution in Theorem 5 is identical to the lifted ILC update (16), in contrast to the non-lifted approach in [14] which minimizes an individual cost function at each sample and as such is fundamentally different. The solution to discrete-time LQT problems with  $S_j^k \succeq 0$ ,  $R_j^k \succ 0$  is well-known, see [18, Section 4.4] or [19] for the situation with direct feedthrough. A Hamiltonian system is defined, leading to a two-point boundary value problem to which a sweep method is applied. For cross-coupled ILC, define

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

The optimal input is  $f_{j+1}(k) = f_j(k) + \Delta f_{j+1}(k)$ , where

$$\begin{aligned} \Delta f_{j+1}(k) = & -\bar{R}_j^{-1}(k) \bar{P}_j(k+1) \Delta x_{j+1}(k) \\ & + \bar{R}_j^{-1}(k) \bar{D}^T S_j^k r_j(k) + \bar{R}_j^{-1}(k) B^T v(k+1), \end{aligned} \quad (33)$$

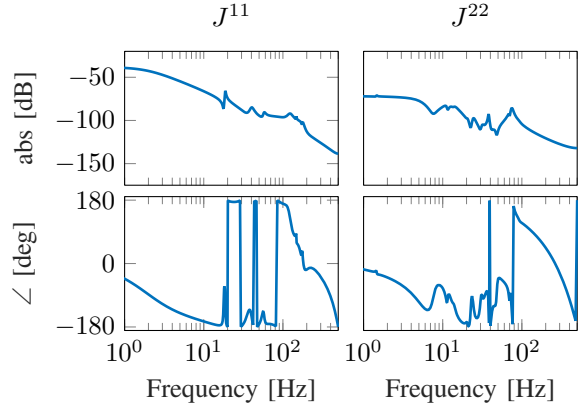


Fig. 3: Bode diagram of the process sensitivity of a decoupled industrial flatbed printer. Input and output 1 correspond to the y-axis, input and output 2 correspond to the x-axis.

$$\text{with } \bar{R}_j(k) = R_j^k + \bar{D}^T S_j^k \bar{D} + B^T P_j(k+1) B, \quad (34)$$

$$\bar{P}_j(k+1) = B^T P_j(k+1) A + \bar{D}^T (S_j^k)^T \bar{C}. \quad (35)$$

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

$$P_j(k) = A^T P_j(k+1) A + \bar{C}^T S_j^k \bar{C} - \quad (36)$$

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

$$\begin{aligned} v_j(k) = & -(\bar{P}_j^T(k+1) \bar{R}_j^{-1}(k) \bar{D}^T - \bar{C}^T) S_j^k r_j(k) \\ & - (\bar{P}_j^T(k+1) \bar{R}_j^{-1}(k) B^T - A^T) v_j(k+1), \end{aligned} \quad (37)$$

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

$$P_j(N) = \bar{C}^T S_j^N \bar{C}, \quad (38)$$

$$v_j(N) = \bar{C}^T S_j^N (\bar{D} \Delta f_{j+1} - r_j(N)). \quad (39)$$

Compared to Theorem 3, Theorem 5 also requires  $W_{\Delta fc,j} \succ 0$ . This is not limiting in practice, as  $W_{\Delta fc,j} \succ 0$  is also required to limit the amplification of iteration-varying disturbances [20]. A similar low-order solution to ILC, which omits the explicit formulation of the ILC update law as an LQT problem, is applied to the specific cases of ILC for intersample behavior in [21] and norm-optimal ILC in [13].

## VI. EXAMPLE

In this section, the proposed cross-coupled ILC framework is illustrated using simulations of an industrial flatbed printer, the process sensitivity of which is shown in Fig. 3. Cross-coupled ILC is applied to this system with coupling matrices  $C_{e,j} = C_{f,j}$  and coupling gains representing the exact contour and tangential errors according to (13). The time- and iteration-invariant weights are given by

$$\begin{aligned} W_{e,j}^k &= \begin{bmatrix} 1.5 & 0 \\ 0 & 0.5 \end{bmatrix}, \quad W_{f,j}^k = \begin{bmatrix} 10^{-10} & 0 \\ 0 & 10^{-9} \end{bmatrix}, \\ W_{\Delta f,j}^k &= 10^{-10} I^{2 \times 2}, \quad \forall k, j. \end{aligned}$$

Thus, the weight on the contour error  $\varepsilon_c$  is higher than that on the tangential error  $\varepsilon_t$ , and the input penalty in the direction of the contour error is smaller, reflecting the aim of minimizing the contour error in this application.

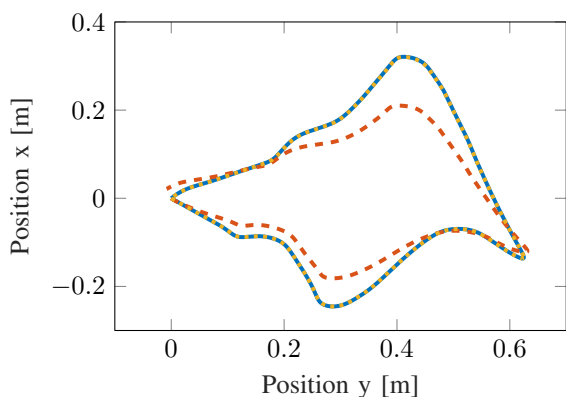


Fig. 4: Reference (—), position output without feedforward (---) and position output with a significantly reduced contour error after ten cross-coupled ILC iterations (.....).

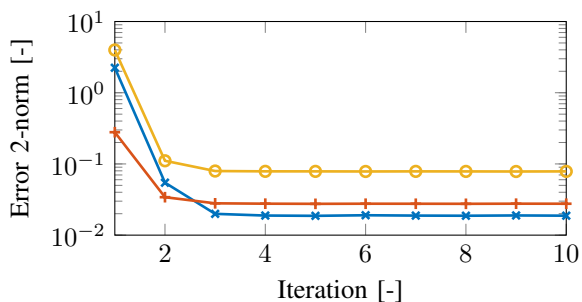


Fig. 5: Convergence of the 2-norm of the contour error (—) and the errors of the individual axes  $y$  (—) and  $x$  (—).

The ILC algorithm with iteration-varying cross-coupling matrices for exact contour errors converges, see Fig. 4 and 5. In this case, the main contour error reduction comes from reducing the error in  $x$ -direction, see also the individual axes errors in Fig. 5. Comparisons with other norm-optimal ILC cost functions are omitted, because different weights lead to different trade-offs between control input and error, making such comparisons completely arbitrary. The advantage of the proposed framework is in the use of exact contour errors, which ensures that the cost function represents the aim of contour tracking, and in the intuitiveness of tuning.

## VII. CONCLUSION

In this paper a new framework for cross-coupled iterative learning control is developed that enables the use of exact contour errors and time- and iteration-varying weights. Conditions for the monotonic convergence of the ILC algorithm are given. In addition, the ILC update law is reframed as a linear quadratic tracking problem, which can be solved efficiently for arbitrary long reference signals. The approach is illustrated on a simulation example of a flatbed printer; experimental results that confirm the theoretical and simulation results are omitted due to space limitations. Future work includes developing design criteria for time-varying weights.

## ACKNOWLEDGMENT

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.

## REFERENCES

- [1] Z.-y. Jia, J.-w. Ma, D.-n. Song, F.-j. Wang, and W. Liu, "A review of contouring-error reduction method in multi-axis CNC machining," *Int. J. Mach. Tools Manuf.*, vol. 125, pp. 34–54, 2018.
- [2] S. Yang, A. H. Ghasemi, X. Lu, and C. E. Okwudire, "Pre-compensation of servo contour errors using a model predictive control framework," *Int. J. Mach. Tools Manuf.*, vol. 98, pp. 50–60, 2015.
- [3] Y. Koren, "Cross-coupled biaxial computer control for manufacturing systems," *J. Dyn. Syst. Meas. Control*, vol. 102, pp. 265–272, 1980.
- [4] F. Huo and A. N. Poo, "Improving contouring accuracy by using generalized cross-coupled control," *Int. J. Mach. Tools Manuf.*, vol. 63, pp. 49–57, 2012.
- [5] Y. T. Shih, C. S. Chen, and A. C. Lee, "A novel cross-coupling control design for bi-axis motion," *Int. J. Mach. Tools Manuf.*, vol. 42, no. 14, pp. 1539–1548, 2002.
- [6] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Syst.*, vol. 26, no. 3, pp. 96–114, 2006.
- [7] L. Blanken and T. Oomen, "Multivariable iterative learning control design procedures: From decentralized to centralized, illustrated on an industrial printer," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 4, pp. 1534–1541, 2020.
- [8] S. Gunnarsson and M. Norrlöf, "On the design of ILC algorithms using optimization," *Automatica*, vol. 37, no. 12, pp. 2011–2016, 2001.
- [9] D. H. Owens, *Iterative Learning Control: An Optimization Paradigm*. London: Springer-Verlag, 2016.
- [10] H. R. Chen, M. Y. Cheng, C. H. Wu, and K. H. Su, "Real time parameter based contour error estimation algorithms for free form contour following," *Int. J. Mach. Tools Manuf.*, vol. 102, 2016.
- [11] K. Barton and A. Alleyne, "A cross-coupled iterative learning control design for precision motion control," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 6, pp. 1218–1231, 2008.
- [12] K. L. Barton and A. G. Alleyne, "A norm optimal approach to time-varying ILC with application to a multi-axis robotic testbed," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 1, pp. 166–180, 2011.
- [13] J. van Zundert, J. Bolder, S. Koekebakker, and T. Oomen, "Resource-efficient ILC for LTI/LTV systems through LQ tracking and stable inversion: Enabling large feedforward tasks on a position-dependent printer," *Mechatronics*, vol. 38, pp. 76–90, 2016.
- [14] H. Sun and A. G. Alleyne, "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, pp. 2046–2051.
- [15] E. Kreyszig, *Introductory Functional Analysis with Applications*. John Wiley & Sons, Inc., 1978.
- [16] N. Stribosch and T. Oomen, "Iterative learning control for intermittently sampled data: Monotonic convergence, design, and applications," *Automatica*, vol. 139, p. 110171, 2022.
- [17] —, "Beyond quantization in iterative learning control: Exploiting time-varying time-stamps," in *Proc. Am. Control Conf.*, 2019, pp. 2984–2989.
- [18] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, "Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers," *IEEE Control Syst. Mag.*, vol. 32, no. 6, pp. 76–105, 2012.
- [19] F. Ebrahimzadeh, J. S. H. Tsai, M. C. Chung, Y. T. Liao, S. M. Guo, L. S. Shieh, and L. Wang, "A generalised optimal linear quadratic tracker with universal applications. Part 2: discrete-time systems," *Int. J. Syst. Sci.*, vol. 48, no. 2, pp. 397–416, 2017.
- [20] T. Oomen and C. R. Rojas, "Sparse iterative learning control with application to a wafer stage: Achieving performance, resource efficiency, and task flexibility," *Mechatronics*, vol. 47, pp. 134–147, 2017.
- [21] T. Oomen, J. van de Wijdeven, and O. H. Bosgra, "System identification and low-order optimal control of intersample behavior in ILC," *IEEE Trans. Automat. Contr.*, vol. 56, no. 11, pp. 2734–2739, 2011.