Experimental Validation of Functional Iterative Learning Control on a One-Link Flexible Arm

Sjoerd Drost[1], Pietro Pustina[2], Franco Angelini[3], Alessandro De Luca[2], Gerwin Smit[1], Cosimo Della Santina[1,4]

***Abstract*— Performing precise, repetitive motions is essential in many robotic and automation systems. Iterative learning control (ILC) allows determining the necessary control command by using a very rough system model to speed up the process. Functional iterative learning control is a novel technique that promises to solve several limitations of classic ILC. It operates by merging the input space into a large functional space, resulting in an over-determined control task in the iteration domain. In this way, it can deal with systems having more outputs than inputs and accelerate the learning process without resorting to model discretizations. However, the framework lacks so far a validation in experiments. This paper aims to provide such experimental validation in the context of robotics. To this end, we designed and built a one-link flexible arm that is actuated by a stepper motor, which makes the development of an accurate model more challenging and the validation closer to the industrial practice. We provide multiple experimental results across several conditions, proving the feasibility of the method in practice.**

I. Introduction

Iterative Learning Control (ILC) is a control strategy for systems that work in a repetitive mode [1], [2]. By performing consecutive trials of the same task, ILC methods learn a feedforward action that progressively improves the control performance from previous experience. At the cost of little system knowledge and data, these approaches produce simple and robust controllers characterized by quantifiable properties and performance. In addition, ILC can improve feedback controllers by modifying the reference signal only [3]. Thus, it applies also to closed-loop systems, such as industrial robots, without affecting their stability features. For these reasons, the ILC framework has been proposed to control several complex systems, e.g., batch processes [4], [5], robots [6]–[8], and biological systems [9].

The functional approach to ILC has been pioneered in the discrete-time domain in [10]–[12]. These works aim to improve the extrapolation properties of ILC controllers w.r.t. exogenous signals, such as set-point variations, and non-repeating disturbances. More recently, [13] proposes a framework called fILC that leverages functional expansions directly in the continuous-time domain. Among other things, this strategy allows learning continuous control action to perform trajectory tracking for non-square linear systems, i.e., dynamical systems with fewer inputs than outputs. Remarkably, many processes of practical interest, including flexible and soft robots, satellites and aircrafts, fall in this category. The goal of fILC is to learn a feedforward action that annihilates the tracking error at prescribed time instants.

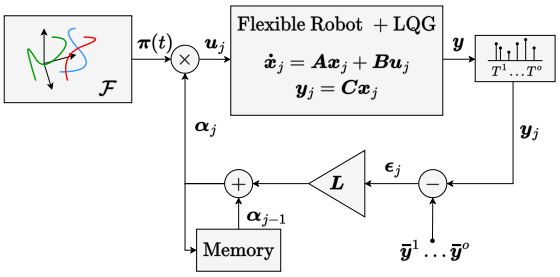


Fig. 1. Schematic representation of the fILC. At every iteration, the controller samples the system output at the desired time instants $\{T^1, \dots, T^o\}$, and computes the output error $\boldsymbol{\epsilon}_j$. Then, $\boldsymbol{\epsilon}_j$ is scaled by a suitable gain matrix $\boldsymbol{L}$, and $\boldsymbol{\alpha}_j$ is updated. Finally, the combination of $\boldsymbol{\alpha}_j$ with the basis functions $\boldsymbol{\pi}(t)$ provides the control action $\boldsymbol{u}_j$.

In this work, we present the first experimental validation of the fILC framework. We test the control law on a one-link flexible arm that we designed and built on purpose. Flexible link robots represent an ideal testbed for fILC. In fact, due to their continuum nature, these robotic systems are intrinsically underactuated [14], [15]. Their control, either in configuration or task space, is challenging and has been the subject of extensive studies over the last forty years [16]–[20]. Notably, several references [21]–[25] proposed ILC for position control and vibration suppression of flexible robots. To render the control more challenging while getting closer to systems commonly used in industry, actuation is performed by a stepper motor. We propose then a strategy for deriving a simple system model to be used as base for control derivations. We first close a loop on the system with a linear quadratic Gaussian (LQG) regulator, and then use fILC to learn a reference for the low-level controller. The control objective is to follow arbitrary motor trajectories while keeping small tip oscillations. This task defines a non-square control problem between one-dimensional input to the motor and the two-dimensional controlled output. The experimental results demonstrate that the fILC is a valid strategy for controlling flexible link robots.

In summary, this paper contributes to the state of the art in ILC and control of flexible link robots by:

- designing and fabricating a new robotic testbed made of a flexible link actuated via a stepper motor;
- proposing a simple yet effective model for such system;

[1]Department of Cognitive Robotics, Delft University of Technology, Delft, The Netherlands. [2]Department of Computer, Control and Management Engineering, Sapienza University of Rome, Rome, Italy. [3]Research Center "Enrico Piaggio" and Department of Information Engineering, University of Pisa, Pisa, Italy. [4]Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Oberpfaffenhofen, Germany. Emails: s.c.drost@student.tudelft.nl, {pustina, deluca}@diag.uniroma1.it, franco.angelini@unipi.it, {g.smit, c.dellasantina}@tudelft.nl. Financial support from PNRR MUR project PE0000013-FAIR is acknowledged.

- providing extensive experimental validation to the fILC framework.

II. Functional Iterative Learning Control

This section briefly reviews fILC (see Fig. 1). Please refer to [13] for an in-depth introduction to the framework. Consider the continuous-time linear dynamics

$$\dot{\boldsymbol{x}}_j = \boldsymbol{A}\boldsymbol{x}_j + \boldsymbol{B}\boldsymbol{u}_j, \qquad \boldsymbol{y}_j = \boldsymbol{C}\boldsymbol{x}_j, \tag{1}$$

with $j \in \mathbb{N}_+$ as iteration index, and matrices $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B} \in \mathbb{R}^{n\times l}$, and $\boldsymbol{C} \in \mathbb{R}^{m\times n}$. Assume that the system is non-square, i.e., $l < m$. Given a finite set of o time instants $\{T^1, \dots, T^o\}$, the fILC calculates $\boldsymbol{u}_j$ so that the sampled output $\boldsymbol{y}_j(T^k)$ tracks a desired value $\bar{\boldsymbol{y}}^k$, $\forall k \in \{1, \dots, o\}$, as j goes to infinity. This is accomplished by regarding $\boldsymbol{u}_j$ as an element of a functional space $\mathcal{F}$, i.e., $\boldsymbol{u}_j = \boldsymbol{\pi}(t)\boldsymbol{\alpha}_j$, where $\boldsymbol{\pi}(t) \in \mathbb{R}^{l\times mo}$ is a matrix whose columns are a basis of $\mathcal{F}$, and $\boldsymbol{\alpha}_j \in \mathbb{R}^{mo}$ collects the coefficients representing $\boldsymbol{u}_j$ in $\mathcal{F}$. Then, $\boldsymbol{\alpha}_j$ is updated through the law

$$\boldsymbol{\alpha}_{j+1} = \boldsymbol{\alpha}_j + \boldsymbol{L}\boldsymbol{\epsilon}_j, \tag{2}$$

where $\boldsymbol{L} \in \mathbb{R}^{mo\times mo}$ is a suitable gain matrix that guarantees contraction of the tracking error

$$\boldsymbol{\epsilon}_j = \left(\bar{\boldsymbol{y}}^{1^\top} - \boldsymbol{y}_j^\top(T^1), \ \dots \ \bar{\boldsymbol{y}}^{o^\top} - \boldsymbol{y}_j^\top(T^o)\right)^\top \in \mathbb{R}^{mo}.$$

III. Experimental Design

To experimentally validate the fILC, a non-square linear system is required. Flexible link robots lie in this class of systems. Indeed, they are usually modeled as linear systems [26] and present a single input to control their whole state. For these reasons, we designed and built the one-link flexible arm shown in Fig. 2. The platform consists of a carbon fiber beam clamped to a stepper motor, whose model is 17HS24-2104D-E1000. A DRV8825 driver is used to drive the stepper motor. A MPU9250 gyroscope measures the tip angular velocity on 3 axes with a frequency of 1 [MHz]. Lightweight 3D printed parts fix in place the arm and the gyroscope. Finally, thin copper wires connect the gyroscope to a ESP32 microprocessor which sends sensor and motor data to a workstation running Simulink. The workstation computes the control law and sends the command back to the microprocessor for execution. The control loop runs at 500 [Hz]. We built the link to be thin and slender, $(\mathrm{H} \times \mathrm{W} \times \mathrm{L}) = (10 \times 0.5 \times 320) \cdot 10^{-3}$[m], so as to result in large oscillations. The hub radius and its weight are $25 \cdot 10^{-3}$[m] and $2.6 \cdot 10^{-3}$[kg], respectively. A slender profile, in combination with the low material damping ratio, produces structural vibrations with large amplitude in the plane of motion. Figure 3 shows the evolution of the motor position and tip angular velocity in response to an open-loop step input of 0.5 [rad]. The robot end oscillates for more than 10 [s] and, because of the coupling effects, also the motor vibrates around the commanded position. Figure 4 shows a photo sequence of the robot motion in the time window $t \in [2; 2.5]$ [s].

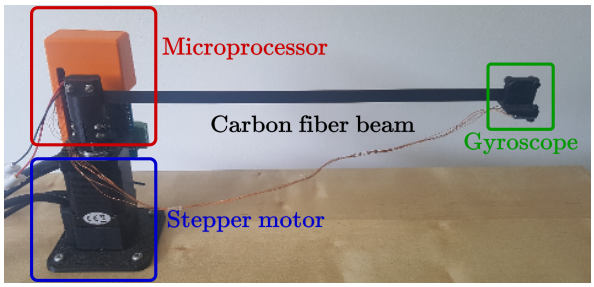


Fig. 2. The one-link flexible arm designed and built in this work. The platform consists of a stepper motor, a slender and flexible carbon fiber link, and a gyroscope measuring the angular velocity of its tip. A microprocessor in the orange box exchanges sensor and motor data with a workstation.

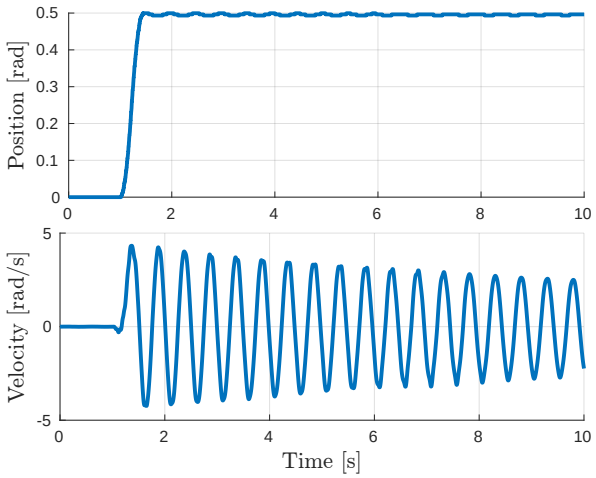


Fig. 3. Open-loop time evolution of motor position and tip angular velocity for a step command of 0.5 [rad] applied at $t = 1$ [s]. The motor and the tip oscillate for more than 10 [s] without a proper control action. Note also that the robot traces two cycles every second.

IV. Dynamic Model and Low Level Control

A. Beam dynamics

Consider a one-link flexible robot with length L moving on a horizontal plane, i.e., without gravity. Under the hypothesis of small deformations, the transverse position y of a point at a distance $x \in [0; L]$ from the base along the neutral axis is $y(t, x) = x\theta(t) + w(t, x)$, where $\theta(t)$ is the motor angle and $w(t, x)$ the lateral beam deflection [26]. To cope with the distributed nature of the above equation, we approximate $w(t, x)$ following the finite element method of [26]. The beam is partitioned in two segments with mass concentrated at the end of each segment, as illustrated in Fig. 5. Each element has only two degrees of freedom, and its configuration variables are the elastic deflection $w_i(t)$ and the angle $\phi_i(t)$ at its distal end; $i \in \{1, 2\}$. Defining the normalized variable $s \in [0; 1]$, the longitudinal position of a point belonging to the i-th element is

$$x_i(s) = x_{i-1}(1) + l_i s; \ \ x_0(1) = 0, \tag{3}$$

being l_i its length. Then, the deflection at $x_1(s)$ and $x_2(s)$ takes, respectively, the expression

$$w(t, x_1(s)) = (\lambda_1(t) \ \lambda_2(t))^\top \boldsymbol{\sigma}_1(s), \tag{4}$$

and

$$w(t, x_2(s)) = (\lambda_1(t) \ \lambda_2(t) \ \lambda_3(t) \ \lambda_4(t))^\top \boldsymbol{\sigma}_2(s), \tag{5}$$

where $\boldsymbol{\lambda} := (w_1 \ \phi_1 \ w_2 \ \phi_2)^\top$ and $\boldsymbol{\sigma}_i(s); i \in \{1, 2\}$ are vectors of Hermite polynomials [1]. Thus, the robot kinematics is described by the configuration $\boldsymbol{q} = \begin{pmatrix} \theta & \boldsymbol{\lambda}^\top \end{pmatrix}^\top \in \mathbb{R}^5$. Standard steps in Lagrangian formulation yield the equations

[1] See [26] for more details.

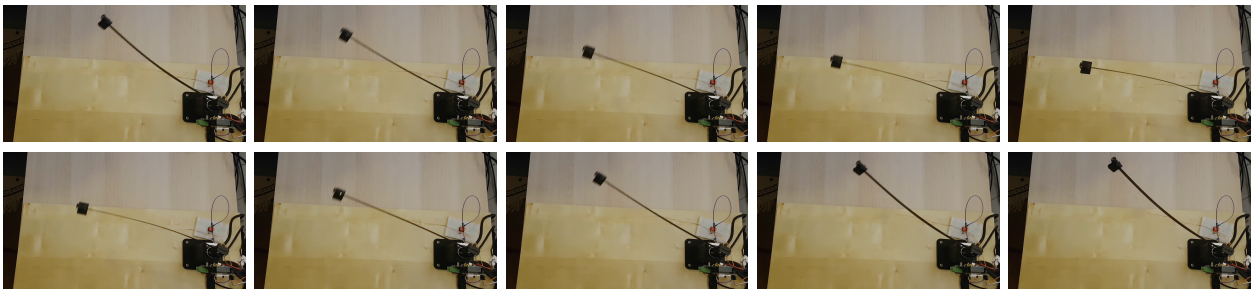


Fig. 4. Photo sequence of the open-loop evolution for a step input of $0.5\,[\mathrm{rad}]$ in the time window $t \in [2; 2.5]$ [s]. In accordance with Fig. 3, this time window corresponds to a complete oscillation period of the robot tip. As expected, due to its slender profile, the beam undergoes large deflections.

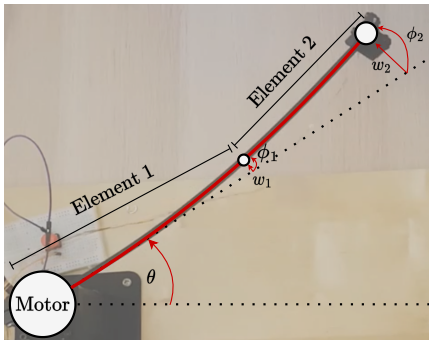


Fig. 5. Schematic representation of the flexible robot. The beam is discretized by introducing two nodes, at its middle point and distal end: thus, the beam is divided into two elements with mass concentrated at the nodes. Being the motion planar, the transverse deflections w_i and the angles ϕ_i, $i = 1, 2$, uniquely locate each node w.r.t. the motor angle θ.

of motion

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{D}\dot{\boldsymbol{q}} + \boldsymbol{K}\boldsymbol{q} = \boldsymbol{f}_{\mathrm{a}}, \tag{6}$$

where

$$\boldsymbol{M} = \begin{pmatrix} I_{\mathrm{m}} & \boldsymbol{m}_{\boldsymbol{\lambda}\theta}^{\top} \\ \boldsymbol{m}_{\boldsymbol{\lambda}\theta} & \boldsymbol{M}_{\boldsymbol{\lambda}\boldsymbol{\lambda}} \end{pmatrix} \succ 0, \tag{7}$$

is the constant symmetric mass tensor with $\boldsymbol{m}_{\boldsymbol{\lambda}\theta} \in \mathbb{R}^4$, including the motor inertia I_{m} and link mass, and the matrices

$$\boldsymbol{D} = \begin{pmatrix} 0 & \boldsymbol{0}_{1\times 4} \\ \boldsymbol{0}_{4\times 1} & \boldsymbol{D}_{\boldsymbol{\lambda}\boldsymbol{\lambda}} \end{pmatrix}, \ \boldsymbol{K} = \begin{pmatrix} 0 & \boldsymbol{0}_{1\times 4} \\ \boldsymbol{0}_{4\times 1} & \boldsymbol{K}_{\boldsymbol{\lambda}\boldsymbol{\lambda}} \end{pmatrix}, \tag{8}$$

with $\boldsymbol{K}_{\boldsymbol{\lambda}\boldsymbol{\lambda}}, \boldsymbol{D}_{\boldsymbol{\lambda}\boldsymbol{\lambda}} \succ 0$, model elastic and dissipative effects, respectively. Finally, $\boldsymbol{f}_{\mathrm{a}} = (1\ \boldsymbol{0}_{1\times 4})^{\top}\tau$ is the generalized force, being τ the motor torque.

B. Motor dynamics and identification

Stepper motors, such as the one used in our experimental setup, result in complex models when using first principles. For control design, we approximate the drive as a second-order mechanical system. In fact, being the electrical dynamics usually much faster than the mechanical one, it is reasonable to neglect it and assume that the motor state is $(\theta, \dot{\theta})^{\top}$. Furthermore, we assume the motor internal dynamics to be dominant and not affected by link vibrations. This yields the following motor model

$$\ddot{\theta} + d_{\mathrm{m}}\dot{\theta} + k_{\mathrm{m}}\theta = g_{\mathrm{m}}v, \tag{9}$$

where $k_{\mathrm{m}}, d_{\mathrm{m}}, g_{\mathrm{m}} > 0$ are constants to be identified and v is the drive input. This model has to be combined with the link dynamics. We do so by replacing the first row of (6) with (9). Furthermore, to keep the inertia tensor symmetric, we substitute the new expression of $\ddot{\theta}$ in the beam dynamics leading to the following grey box model

$$\begin{pmatrix} 1 & \boldsymbol{0}_{1\times 4} \\ \boldsymbol{0}_{4\times 1} & \boldsymbol{M}_{\boldsymbol{\lambda}\boldsymbol{\lambda}} \end{pmatrix}\begin{pmatrix} \ddot{\theta} \\ \ddot{\boldsymbol{\lambda}} \end{pmatrix} + \begin{pmatrix} d_{\mathrm{m}} & \boldsymbol{0}_{1\times 4} \\ -\boldsymbol{m}_{\boldsymbol{\lambda}\theta}d_{\mathrm{m}} & \boldsymbol{D}_{\boldsymbol{\lambda}\boldsymbol{\lambda}} \end{pmatrix}\begin{pmatrix} \dot{\phi} \\ \dot{\boldsymbol{\lambda}} \end{pmatrix} + \begin{pmatrix} k_{\mathrm{m}} & \boldsymbol{0}_{1\times 4} \\ -\boldsymbol{m}_{\boldsymbol{\lambda}\theta}k_{\mathrm{m}} & \boldsymbol{K}_{\boldsymbol{\lambda}\boldsymbol{\lambda}} \end{pmatrix}\begin{pmatrix} \theta \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} g_{\mathrm{m}} \\ -\boldsymbol{m}_{\boldsymbol{\lambda}\theta}g_{\mathrm{m}} \end{pmatrix} v. \tag{10}$$

Thus, the system input is v and the output (measured) variables are the motor position and tip angular velocity, i.e., $\boldsymbol{y} = (\theta\ \ \dot{\phi}_2)^{\top}$. Note that stiffness and damping forces now couple rigid and flexible dynamics. Also, it can be shown that the origin is an asymptotically stable equilibrium of (10).

The motor constants $\boldsymbol{a}_{\mathrm{m}} = (k_{\mathrm{m}}\ d_{\mathrm{m}}\ g_{\mathrm{m}})^{\top}$ have been estimated through a least squares identification experiment by applying a step input of amplitude $\pi/2[\mathrm{rad}]$. With the sampled evolution of θ, also $\dot{\theta}$ and $\ddot{\theta}$ can be computed. Then, (9) yields the regressor

$$\boldsymbol{A}_{\mathrm{m}}\boldsymbol{a}_{\mathrm{m}} = \begin{pmatrix} \theta(t_0) & \dot{\theta}(t_0) & \pi/2 \\ \theta(t_1) & \dot{\theta}(t_1) & \pi/2 \\ \vdots & \vdots & \vdots \\ \theta(t_n) & \dot{\theta}(t_n) & \pi/2 \end{pmatrix}\begin{pmatrix} k_{\mathrm{m}} \\ d_{\mathrm{m}} \\ g_{\mathrm{m}} \end{pmatrix} = \begin{pmatrix} \ddot{\theta}(t_0) \\ \ddot{\theta}(t_1) \\ \vdots \\ \ddot{\theta}(t_n) \end{pmatrix} = \boldsymbol{b}_{\mathrm{m}},$$

which provides the parameters estimate $\hat{\boldsymbol{a}}_{\mathrm{m}} = \boldsymbol{A}_{\mathrm{m}}^{\#}\boldsymbol{b}_{\mathrm{m}}$, where $\boldsymbol{A}_{\mathrm{m}}^{\#}$ is the Moore-Penrose pseudoinverse of $\boldsymbol{A}_{\mathrm{m}}$.

C. LQG controller

We design the fILC on top of a LQG regulator. In this way, we can implicitly evaluate the performance of the fILC w.r.t. an optimal controller. Indeed, at the first iteration, the fILC does not provide any contribution since $\boldsymbol{\alpha}_1 = \boldsymbol{0}$. We design the LQG by minimizing the functional cost

$$J(\boldsymbol{z}, v) = \mathbb{E}\left[\int_0^T \boldsymbol{z}^{\top}(\tau)\boldsymbol{Q}\boldsymbol{z}(\tau) + rv^2(\tau)\mathrm{d}\tau\right], \tag{11}$$

where $\boldsymbol{z} = \left(\boldsymbol{q}^{\top}\ \dot{\boldsymbol{q}}^{\top}\right)^{\top}$ is the system state, and $T > 0$, $\boldsymbol{Q} \in \mathbb{R}^{10\times 10} \geq \boldsymbol{0}$, and $r > 0$ are design parameters. Note that the system remains stable under the LQG since (10) is asymptotically stable at the origin.

V. Experimental Results

In this section, we introduce the main result of this work: an experimental validation of the fILC framework in controlling the experimental setup presented in Sec. III. The objective is to control the motor position to an arbitrary reference $\bar{\theta}^k$ while keeping the tip angular velocity as close as possible to zero, i.e., $\dot{\phi}_2^k = 0\,[\mathrm{rad/s}]$, so as to minimize arm vibrations.

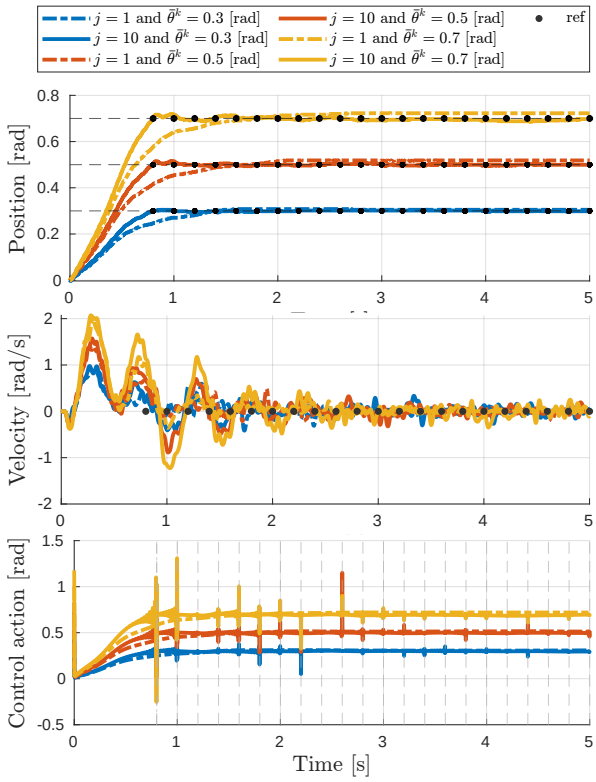


Fig. 6. Experiment 1 (Varying reference angle). Time evolution of the motor position, tip angular velocity, and control input for $\bar{\theta}^k = \bar{\theta} \in \{0.3, 0.5, 0.7\}$ [rad]. The reference is represented with a black dot, whereas solid and dashed lines illustrate each variable at the first (LQG only) and 10−th iteration, respectively.

We design the Kalman filter of the LQG regulator assuming zero mean state and measurement noise with covariance, respectively,

$$\begin{pmatrix} 4.8 \cdot 10^{-5} & \mathbf{0}_{1\times 9} \\ \mathbf{0}_{9\times 1} & 3 \cdot 10^{-2} \cdot \boldsymbol{I}_9 \end{pmatrix}, \begin{pmatrix} 8.88 \cdot 10^{-5} & 0 \\ 0 & 1.77 \cdot 10^{-1} \end{pmatrix}.$$

The LQG gain is computed by taking $T = 5$ [s], $r = 1$, and $\boldsymbol{Q} = \text{diag}\,(1, 0, 0, 0, 5, 1, 0, 0, 0, 1)$. For the fILC, we choose the damped pseudo-inverse as learning factor with damping $\boldsymbol{S} = 10^{-2} \cdot \boldsymbol{I}$, and the base functions of Lemma 1 in [13]. In the first set of experiments, the time instants T^k linked to the reference $\bar{\theta}^k$ are spaced 0.2 [s] apart. We did not observe relevant improvements in choosing smaller values of T^k.

A. Experiment 1 (Varying reference angle)

In this experiment, we command three constant motor angles, i.e., $\bar{\theta}^k = \bar{\theta} \in \{0.3, 0.5, 0.7\}$ [rad]. The fILC acts from $T^1 = 0.8$ [s] on. Figure 6 shows the time evolution of the motor position, tip angular, and control action. At the 10−th iteration, the performance improves significantly with respect to the first iteration (LQG only). Transients dramatically reduce as well as tracking errors. Figure 7 reports the root mean square error (RMSE) with respect to the iteration number. As expected, the error annihilates at the desired time instants. It is important to note that the total velocity of the tip was already low at the first iteration. The goal of the controller is to improve tracking of the angle while keeping the oscillations small. Furthermore, the fILC compensates for model uncertainties by learning a suitable anticipatory action that cancels the steady state error of the LQG. Note also that, after the training phase, the fILC dampens tip vibrations by injecting a suitable oscillatory command.

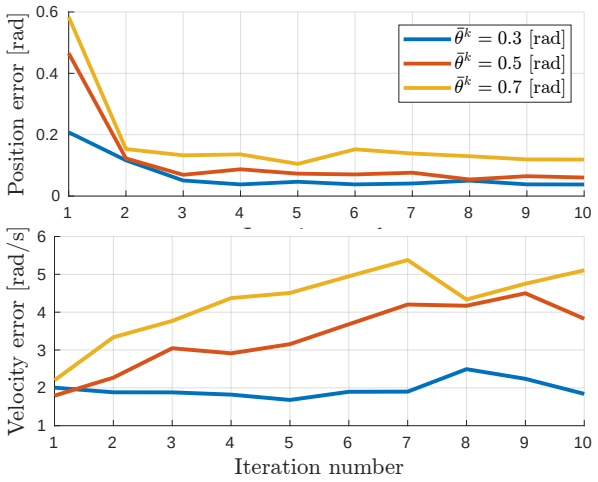


Fig. 7. Experiment 1 (Varying reference angle). Motor and tip RMSE of the closed-loop system for $\bar{\theta}^k = \bar{\theta} \in \{0.3, 0.5, 0.7\}$ [rad] and $\dot{\phi}_2 = 0$ [rad/s].

B. Experiment 2 (Varying start time)

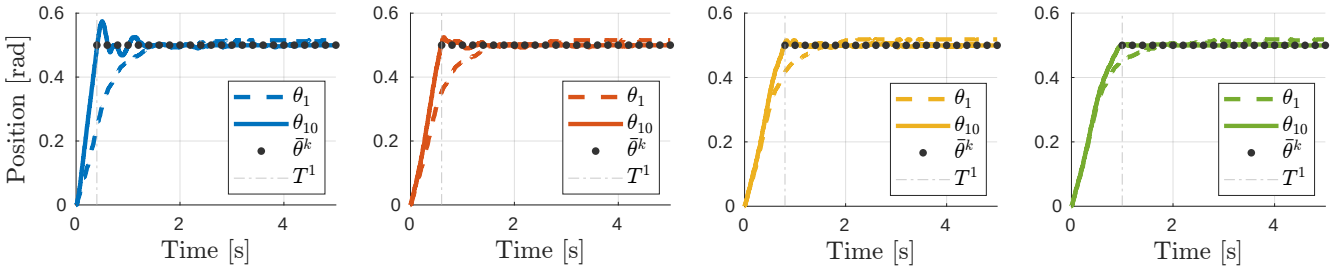
In this second experiment, we evaluate the closed-loop performance by turning on the fILC at different time instants, i.e., $T^1 \in \{0.4, 0.6, 0.8, 1\}$ [s]. The reference motor angle is $\bar{\theta}^k = \bar{\theta} = 0.5$ [rad]. Figures 8 and 9 show, respectively, the closed-loop motor angle and tip velocity as T^1 varies. The controller exhibits performance consistent with the previous experiment. When T^1 increases, the fILC modifies less the optimal trajectory resulting from the LQG because a less aggressive motion is asked. This fact also results from the evolution of the control action in Fig. 10. Furthermore, the response has smaller overshoots, reflecting shorter settling times, as clear also from the RMSE in Fig. 11.

C. Experiment 3 (Repeatability)

We verify the reliability of the fILC by repeating the same experiment twenty times and without updating the fILC control law. We again command $\bar{\theta}^k = \bar{\theta} = 0.5$ [rad] and set $T^1 = 0.8$ [s]. Figure 12 reports the output and input variables over the twenty iterations. The closed-loop evolution is consistent through all the experiment repetitions with small variations due to measurement noise.

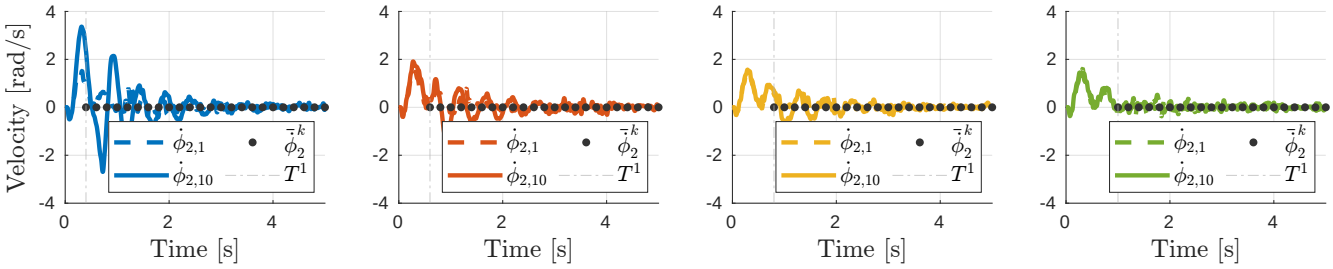
D. Experiment 4 (Less aggressive LQG)

This experiment evaluates the insensitivity of the closed-loop w.r.t. different choices of the LQG gain. Indeed, the fILC should learn a feedforward action that yields similar performance independently of the LQG controller. In particular, we compare the LQG gain used up to now with a less aggressive gain derived by considering $\boldsymbol{Q} = \text{diag}\,(10, 0, 0, 0, 10, 15, 0, 0, 0, 1)$ in (11), i.e., by weighting more the motor velocity. Figure 13 shows the closed-loop evolution of the drive angle, tip angular velocity, and control action. As expected, the fILC learns a suitable feedforward action also when the LQG is less aggressive. However, this comes at the expense of a longer training phase. In fact, at 10−th iteration, the two closed-loops' performance does



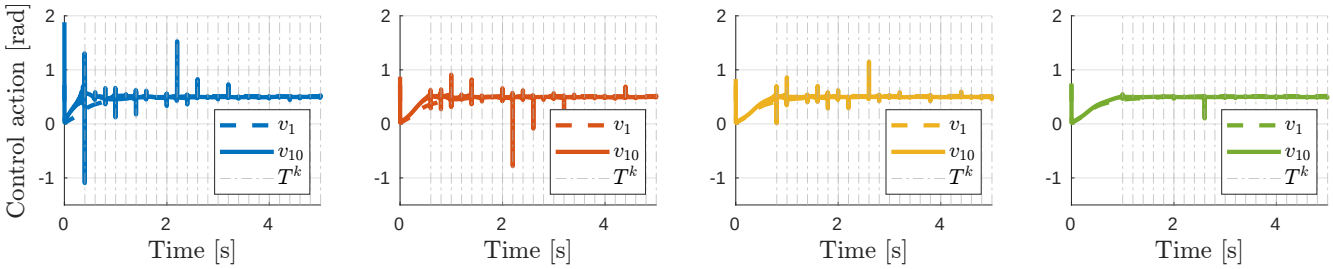
(a) $T^1 = 0.4$ [s] (b) $T^1 = 0.6$ [s] (c) $T^1 = 0.8$ [s] (d) $T^1 = 1$ [s]

Fig. 8. Experiment 2 (Varying starting time). Time evolution of the motor position for $T^1 \in \{0.4, 0.6, 0.8, 1\}$ [s]. The reference $\bar{\theta}^k = \bar{\theta} = 0.5$ [rad] is represented with a black dot, whereas solid and dashed lines illustrate the motor angle at the first (LQG only) and 10−th iteration, respectively. A vertical dashed line shows when the fILC is turned on.



(a) $T^1 = 0.4$ [s] (b) $T^1 = 0.6$ [s] (c) $T^1 = 0.8$ [s] (d) $T^1 = 1$ [s]

Fig. 9. Experiment 2 (Varying starting time). Time evolution of the tip angular velocity for $T^1 \in \{0.4, 0.6, 0.8, 1\}$ [s], a commanded velocity of 0 [rad/s], and $\bar{\theta}^k = \bar{\theta} \in \{0.3, 0.5, 0.7\}$ [rad], organized as in Fig. 8.



(a) $T^1 = 0.4$ [s] (b) $T^1 = 0.6$ [s] (c) $T^1 = 0.8$ [s] (d) $T^1 = 1$ [s]

Fig. 10. Experiment 2 (Varying starting time). Time evolution of the control action for $T^1 \in \{0.4, 0.6, 0.8, 1\}$ [s], a commanded velocity of 0 [rad/s], and $\bar{\theta}^k = \bar{\theta} \in \{0.3, 0.5, 0.7\}$ [rad], organized as in Fig. 8.

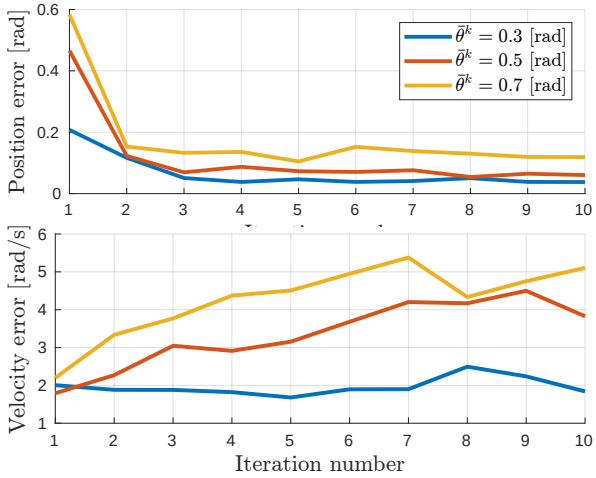


Fig. 11. Experiment 2 (Varying starting time). Closed-loop motor and tip RMSE for $T^1 \in \{0.4, 0.6, 0.8, 1\}$ [s], a commanded velocity of 0 [rad/s], and $\bar{\theta}^k = \bar{\theta} \in \{0.3, 0.5, 0.7\}$ [rad].

not correspond. Nonetheless, the fILC with a less aggressive LQG gain still outperforms the LQG alone. At the end of the training process, the commands of the two loops become similar. This is consistent with the expectation that, as the iterations increase, the control action becomes a pure feedforward independent of the feedback controller used.

E. Experiment 5 (Trajectory tracking)

In this last experiment, we command a time-varying reference consisting of a sequence of two rectangular pulses, having each one amplitude 0.6 [rad], duration 1 [s] and centered in zero, followed by a constant signal of −0.3 [rad], see the yellow line in Fig. 14. Note that, being discontinuous, the trajectory is particularly challenging to follow. The time instants are spaced 0.75 [s] apart and $T^1 = 0.75$ [s]. Figure 14 depicts the time evolution of the motor position, tip angular velocity, and motor input. The closed-loop system exhibits poor performance at the first iteration (LQG only). Indeed, being a pure feedback regulator, the LQG lags behind the

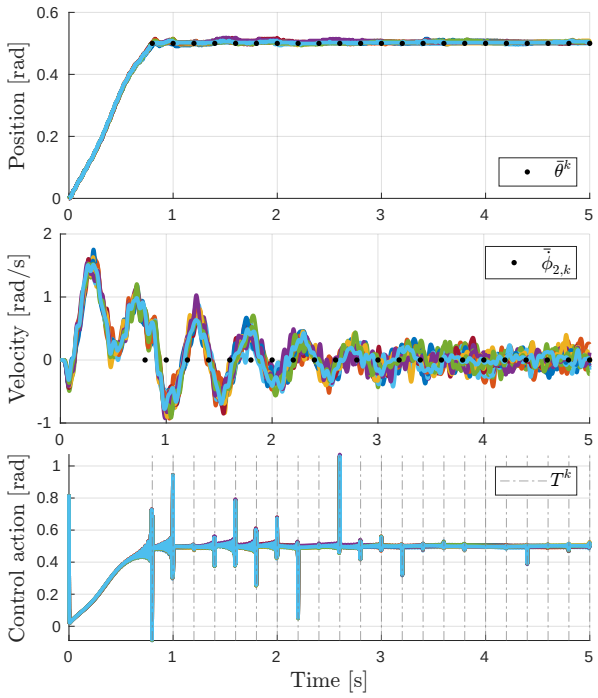


Fig. 12. Experiment 3 (Repeatability). Time evolution of the motor position, tip angular velocity, and control action for $\bar{\theta}^k = \bar{\theta} = 0.5\,[\text{rad}]$, a commanded velocity of $0\,[\text{rad/s}]$, and $T^1 = 0.8\,[\text{s}]$ over twenty repetitions of the same experiment.

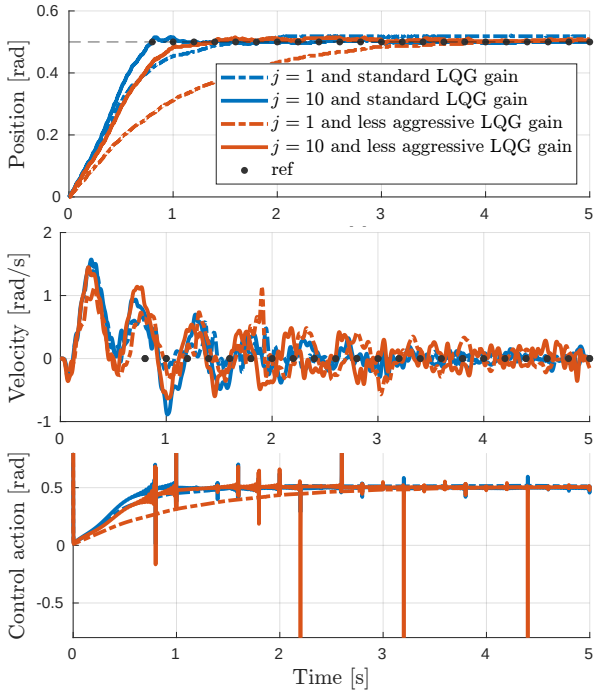


Fig. 13. Experiment 4 (Less aggressive LQG). Time evolution of the motor position, tip angular velocity, and control input for different LQG gains. The motor reference, represented with a black dot, is $\bar{\theta}^k = \bar{\theta} = 0.5\,[\text{rad}]$. Solid and dashed lines illustrate each variable at the first (LQG only) and $10-$th iteration, respectively.

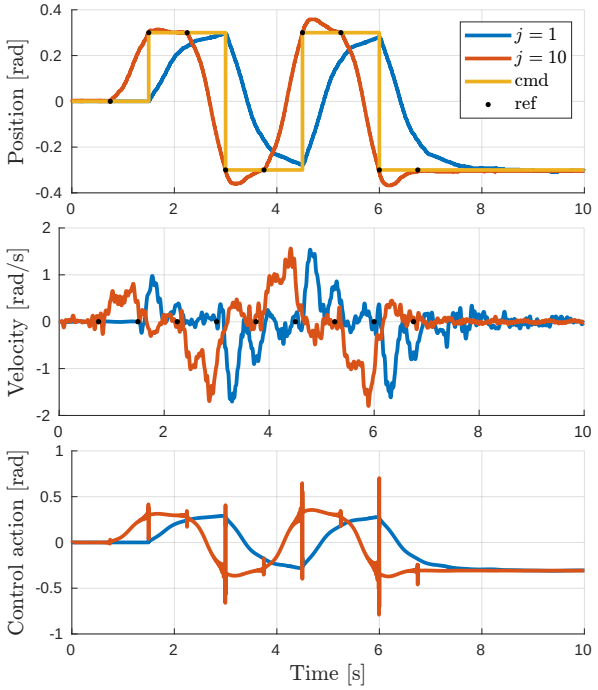


Fig. 14. Experiment 5 (Trajectory tracking). Time evolution of the motor position, tip angular velocity, and control input for the yellow trajectory. In each plot, the blue and red lines show the evolution of the corresponding variable at the first (LQG only) and $10-$th iteration. The fILC reference is represented with a black dot.

command. On the contrary, the fILC learns a suitable anticipatory action, which yields exact tracking of the sampled output in few iterations. Indeed, at the 10-th iteration, the control input leads the motor position forward the reference.

VI. Discussion and Conclusions

In this paper, we proposed the first experimental validation of functional Iterative Learning Control, i.e., a novel Iterative Learning Control framework for trajectory tracking of non-square continuous-time linear systems with sampled outputs. To this end, we designed and built a one-link flexible robot actuated through a stepper motor. We proposed a control-oriented model for the stepper motor, which naturally fits the dynamic formulation.

Extensive experimental results demonstrate the beneficial effects of fILC. The proposed method can execute both regulation (Exp. 1-4) and tracking (Exp. 5) tasks, learning the suitable control input in a limited number of iterations (Figs. 7 and 11). In all experiments, the method learns a suitable anticipatory action to dampen out the tip oscillations and compensate for model inaccuracies. The proposed technique is also robust and reliable, achieving analogous results in 20 repetitions of the same task (Exp. 3).

The fILC input at the first iteration equals zero in each experiment. This means that the performance of any first trial corresponds to the performance of the LQG controller alone. As shown in Figs. 6, 8 and 9, the sole LQG control law is not sufficient to execute the desired motions. On the other hand, the primary influence of the LQG on the fILC stands on the learning rate: reducing the LQG gain leads to longer learning phases without affecting the actual fILC performance (Exp. 4).

REFERENCES

[1] G. Casalino and G. Bartolini, "A learning procedure for the control of movements of robotic manipulators," *IASTED Symp. on Robotics and Automation*, 1984.

[2] K. L. Moore, *Iterative Learning Control for Deterministic Systems*. Springer Science & Business Media, 2012.

[3] R. Chi, H. Li, D. Shen, Z. Hou, and B. Huang, "Enhanced P-type control: Indirect adaptive learning from set-point updates," *IEEE Trans. on Automatic Control*, 2022.

[4] S. Hao, T. Liu, W. Paszke, and K. Galkowski, "Robust iterative learning control for batch processes with input delay subject to time-varying uncertainties," *IET Control Theory & Applications*, vol. 10, no. 15, pp. 1904–1915, 2016.

[5] J. Lu, Z. Cao, R. Zhang, and F. Gao, "Nonlinear monotonically convergent iterative learning control for batch processes," *IEEE Trans. on Industrial Electronics*, vol. 65, no. 7, pp. 5826–5836, 2017.

[6] E. L. Sauser, B. D. Argall, G. Metta, and A. G. Billard, "Iterative learning of grasp adaptation through human corrections," *Robotics and Autonomous Systems*, vol. 60, no. 1, pp. 55–71, 2012.

[7] W. Chen and M. Tomizuka, "Dual-stage iterative learning control for MIMO mismatched system with application to robots with joint elasticity," *IEEE Trans. on Control Systems Technology*, vol. 22, no. 4, pp. 1350–1361, 2013.

[8] X. Li, Y.-H. Liu, and H. Yu, "Iterative learning impedance control for rehabilitation robots driven by series elastic actuators," *Automatica*, vol. 90, pp. 1–7, 2018.

[9] C. T. Freeman, "Upper limb electrical stimulation using input-output linearization and iterative learning control," *IEEE Trans. on Control Systems Technology*, vol. 23, no. 4, pp. 1546–1554, 2014.

[10] J. van de Wijdeven and O. H. Bosgra, "Using basis functions in iterative learning control: Analysis and design theory," *Int. J. of Control*, vol. 83, no. 4, pp. 661–675, 2010.

[11] J. Bolder and T. Oomen, "Rational basis functions in iterative learning control—with experimental verification on a motion system," *IEEE Trans. on Control Systems Technology*, vol. 23, no. 2, pp. 722–729, 2014.

[12] L. Blanken, G. Isil, S. Koekebakker, and T. Oomen, "Flexible ILC: Towards a convex approach for non-causal rational basis functions," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12 107–12 112, 2017.

[13] C. Della Santina and F. Angelini, "Iterative learning in functional space for non-square linear systems," in *Proc. 60th IEEE Conf. on Decision and Control*, 2021, pp. 5858–5863.

[14] A. De Luca, "Flexible robots," in *Encyclopedia of Systems and Control*. Springer, 2020, pp. 1–9. [Online]. Available: http://link.springer.com/10.1007/978-1-4471-5102-9_176-3

[15] C. Della Santina, "Flexible manipulators," in *Encyclopedia of Robotics*. Springer, pp. 1–15. [Online]. Available: https://link.springer.com/10.1007/978-3-642-41610-1_182-1

[16] M. W. Spong, "On the force control problem for flexible joint manipulators," *IEEE Trans. on Automatic Control*, vol. 34, no. 1, pp. 107–111, 1989.

[17] L. Zollo, B. Siciliano, A. De Luca, and E. Guglielmelli, "PD control with on-line gravity compensation for robots with flexible links," in *Proc. European Control Conf.*, 2007, pp. 4365–4370.

[18] E. Pereira, J. R. Trapero, I. M. Díaz, and V. Feliu, "Adaptive input shaping for single-link flexible manipulators using an algebraic identification," *Control Engineering Practice*, vol. 20, no. 2, pp. 138–147, 2012.

[19] T. Meng and W. He, "Iterative learning control of a robotic arm experiment platform with input constraint," *IEEE Trans. on Industrial Electronics*, vol. 65, no. 1, pp. 664–672, 2018.

[20] A. Cristofaro, A. De Luca, and L. Lanari, "Linear-quadratic optimal boundary control of a one-link flexible arm," *IEEE Control Systems Lett.*, vol. 5, no. 3, pp. 833–838, 2021.

[21] M. Poloni and G. Ulivi, "Iterative learning control of a one-link flexible manipulator," in *Proc. 3rd IFAC Symp. on Robot Control*, 1991, pp. 393–398.

[22] P. Lucibello, S. Panzieri, and G. Ulivi, "Repositioning control of a two-link flexible arm by learning," *Automatica*, vol. 33, no. 4, pp. 579–590, 1997.

[23] J. O. Pedro and R. V. Smith, "Real-time hybrid PID/ILC control of two-link flexible manipulators," *IFAC-PapersOnLine*, vol. 50, no. 2, pp. 145–150, 2017.

[24] X. Xing and J. Liu, "Modeling and robust adaptive iterative learning control of a vehicle-based flexible manipulator with uncertainties," *Int. J. of Robust and Nonlinear Control*, vol. 29, no. 8, pp. 2385–2405, 2019.

[25] T. Meng and W. He, *Iterative Learning Control for Flexible Structures*. Springer, 2020.

[26] B. Chapnik, G. Heppler, and J. Aplevich, "Modeling impact on a one-link flexible robotic arm," *IEEE Trans. on Robotics and Automation*, vol. 7, no. 4, pp. 479–488, 1991.