2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

A Simple Computationally Efficient Path ILC for Industrial Robotic Manipulators

Michael Schwegel[1] and Andreas Kugi[1,2]

Abstract— **In this paper, a numerically efficient flexible control scheme for the absolute accuracy of industrial robots is presented and experimentally validated. A model-based controller that leverages all typically available parameters is combined with an online path iterative learning controller (ILC). The ILC law is employed to compensate for the unknown residual error dynamics caused by elastic and transmission effects. The proposed approach combines several benefits, including the possibility of a continuous execution of trials, a straightforward generalization of the learned data to different execution speeds, and learning from partial trials. The experimental validations on a 6-axis industrial robot with a laser tracker absolute measurement system show a 95% improvement in absolute accuracy after two trials. When the laser tracker is removed, the learned feedforward controller can sustain the accuracy achieved even without trial-by-trial learning.**

I. INTRODUCTION

Industrial robots are increasingly used in applications that require a high accuracy such as additive manufacturing, precision machining, and medical robotics. While state-of-the-art model-based robot control strategies exhibit a high fidelity in compensating known dynamical effects, achieving sub-millimeter accuracy for dynamical movements of industrial robots remains a challenging task. The main sources of errors comprise joint elasticities, the mismatched control input, friction effects, transmission errors, and kinematic errors due to tolerances, and wear, see, e.g., [1], [2], [3]. Iterative learning control (ILC) can eliminate unknown, but repetitive effects by recursively improving the accuracy of the robotic manipulator while traversing a path repeatedly. To improve the absolute accuracy of an industrial robot, a model-based controller is augmented with an ILC approach in this paper.

Absolute accuracy in task-space is crucial for industrial robots in demanding applications, such as advanced machining tasks, e.g., [4], [5], medical applications, and surgery robotics, e.g., [6]. Despite the importance of task-space accuracy, this topic has gained little attention in the literature compared to joint-space precision and approaches based on joint encoders, see, e.g., [7], [8]. However, it is well known that significant dynamics of the robot links cannot be observed using motor-side encoders, see, e.g., [9]. Hence, this paper uses the full system knowledge in a computed-torque controller with feedforward compensation based on a mathematical robot model to achieve accurate task-space control. The controller leverages all typically available parameters of an industrial robot. Additionally, a cascaded velocity feedback controller is employed to counteract non repetitive errors. To further improve the accuracy of the control concept, an efficient ILC scheme is employed to compensate for unmodeled effects such as parameter uncertainties, transmission error dynamics, and couplings between elastic and friction effects. This is in contrast to many existing works, where ILC is used to substitute for a model-based feedforward control, see, e.g., [10], [11]. Liu et al. [12] use a pure feedforward inverse-dynamics controller and a joint space precision approach. In this paper, a full computed-torque is employed and since the reference path is modified by the ILC, the presented control law can be applied easily to other control architectures. A laser tracker system is employed to measure the position of the tool center point (TCP) of the robot. The orientation of the robot is measured by the motor encoders.

The transmission error dynamics, which are the main focus of this work, exhibit a nonlinear behavior and change drastically throughout the workspace. For example, imperfections of a gear tooth, can influence the end-effector motion up to a hundred times per joint revolution, see, e.g., [13]. Similarly, the coupling of gravitational load, elasticities, and friction effects causes a significant hysteresis and changes the observed damping for movements of a link with or against gravity. In contrast to [9], where unknown weights serve as a basis for a local learning model, we investigate trajectories spanning a large part of the workspace, and all available information about the process is used. Therefore, the remaining error dynamics are challenging to model, and hence, a simple data-driven proportional-derivative (PD) ILC learning law is employed to cope with the unknown dynamics.

An additional benefit of the proposed PD ILC law is that it does not depend on the complete trajectory and can be applied recursively, hence realizing an online ILC approach. In the literature, online ILC approaches like [14] use a linear error model, which does not reflect the nonlinear effects of the investigated error dynamics. In [15], an online approach for learning dynamic motion primitives is developed, and [16] presents a force learning task.

In order to achieve the desired absolute accuracy, the robot end-effector position must be measured with high accuracy at a high rate. In the industry, laser trackers are typically used to calibrate robots to improve accuracy. Due to the high price of a laser tracker, the ILC inputs are pre-recorded and then

We thank KEBA Industrial Automation GmbH for their financial and intellectual support of this project.

[1]Michael Schwegel and Andreas Kugi are with the Complex Dynamical Systems Group, Automation and Control Institute (ACIN), TU Wien, 1040 Vienna, Austria michael.schwegel@tuwien.ac.at

[2]Andreas Kugi is with AIT, Austrian Institute of Technology GmbH, 1210 Vienna, Austria andreas.kugi@ait.ac.at

used as a feedforward input. Therefore, the laser tracker is only needed in the initial learning phase. In order to be able to apply the pre-recorded feedforward signals to different execution speeds of the robot trajectory, the path parameter of the reference curve serves as an index variable. Different time-scaling approaches that rely on a system model have been reported in the literature, see, e.g., [17], [18], [19]. Compared to these approaches, faced with complex error dynamics, we take a simple path parameter ILC approach that allows us to scale the execution speed of the reference path. Phase indexing approaches are reported in the literature, see, e.g., [20], [21].

Combining the online ILC with the path parameter indexing alleviates two classical ILC assumptions: the requirement of fixed-length trials and the fixed starting point for each trial. In contrast to prior works, the combination of parameter indexing and online learning allows the proposed ILC to be applied to variable speed traversals, partially traversed paths, continuously executed trials, i.e., repetitive control problems, e.g., [22].

The scientific contribution of this paper can be summarized as follows: This paper presents a simple, efficient, and flexible control scheme with learning control for the absolute accuracy of an industrial robot manipulator. To fully leverage the ILC, the governing dynamics of the robot are compensated by a computed-torque and feedforward controller. The proposed ILC law can supplement the performance of existing control strategies with little requirements on the computational hardware and control structure.

In contrast to many prior investigations, this work focuses on the unknown transmission error dynamics and uses of all typically available parameters and a detailed robot model. Furthermore, the absolute accuracy of the robot, which is measured using a laser tracker, is the main focus of this work. Since the laser tracker may only be available for an initial learning phase, the efficacy of the learned ILC signals as feedforward trajectories is investigated. The proposed ILC approach is also applied to improve the tool orientation accuracy of the robot with motor encoder measurements only.

Additionally, the approach features high flexibility and ease of practical application. All calculations can be done recursively, and a continuous learning phase is possible even with variations in the reference trajectory speed. Further, the classical requirement of strictly repeating trials for the ILC is softened. Other works in the field, such as [9], use multiple filters that must be tuned after each trial. Here, we only use a single straightforward learning approach that does not require any intervention between the trials and achieves a comparable accuracy for a complex trajectory on industrial robot.

II. Problem Statement

This section aims to improve the absolute positioning accuracy of an industrial robot for a repeating trajectory by recursively learning unmodeled effects. The ILC approach used to learn these effects does not make assumptions about the employed control approach or the knowledge of robot parameters. To improve the absolute accuracy of the robot, a laser tracker measures the absolute position of the tool center point (TCP) of the robot. The experimental validation was done with the 6 degrees-of-freedom industrial robot *Comau Racer 7-1.4* and the laser tracker *Leica Absolute Tracker AT960*, depicted in Fig 1.

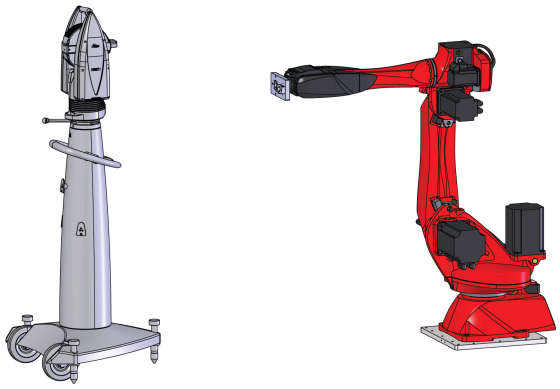

Fig. 1. Industrial robot *Comau Racer 7-1.4* and *Leica Absolute Tracker AT960*.

A. Mathematical Model

The mathematical model of the rigid-body robot is given by

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = -\boldsymbol{\tau}_f + \boldsymbol{\tau}_c, \tag{1}$$

where $\mathbf{M}(\mathbf{q})$ is the mass matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$, the Coriolis matrix, and $\mathbf{g}(\mathbf{q})$ the vector of gravitational torques. Furthermore, $\boldsymbol{\tau}_f$ represents the torques caused by friction effects, and $\boldsymbol{\tau}_c$ are the control input torques on the robot joints. The mapping between the joint-space coordinates $\mathbf{q}$ and the Cartesian task-space position $\mathbf{x}$ of the TCP is given by the forward kinematics

$$\mathbf{x} = \mathbf{f}(\mathbf{q}) \,, \tag{2}$$

and the differential kinematics are defined using the manipulator Jacobian matrix $\mathbf{J}_v(\mathbf{q})$ by

$$\dot{\mathbf{x}} = \mathbf{J}_v\dot{\mathbf{q}} \,. \tag{3}$$

III. Control Concept

Let $\mathbf{q}_r, \dot{\mathbf{q}}_r$, and $\ddot{\mathbf{q}}_r$ be a joint reference trajectory and its derivatives. The control input $\boldsymbol{\tau}_c$ consists of three parts

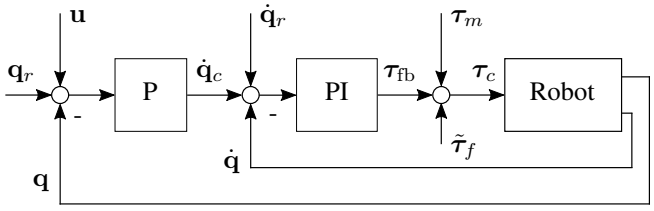

Fig. 2. Flowchart of the robot control structure.

$$\boldsymbol{\tau}_c = \boldsymbol{\tau}_m + \tilde{\boldsymbol{\tau}}_f + \boldsymbol{\tau}_{\text{fb}} \,, \tag{4}$$

with the computed-torque control

$$\boldsymbol{\tau}_m = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}_r + \mathbf{C}(\dot{\mathbf{q}}, \mathbf{q})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) \tag{5}$$

and the feedforward friction compensation term, e.g., [23],

$$\tilde{\boldsymbol{\tau}}_f = \mathbf{K}_v\dot{\mathbf{q}}_r + \mathbf{K}_c\text{sign}(\dot{\mathbf{q}}_r) \; . \tag{6}$$

The entries of the positive diagonal matrices $\mathbf{K}_v$ and $\mathbf{K}_c$ refer to the viscous and Coulomb friction coefficients in the joints, respectively, and were found from measurements.

In order to account for non-repetitive unmodeled effects, like transmission errors and elasticities, a cascaded feedback controller (the term $\boldsymbol{\tau}_{\text{fb}}$ in (4)) is employed with a PI velocity controller in the inner loop and a proportional position controller in the outer loop. Fig. 2 depicts the block diagram of this state-of-the-art control concept. The PI velocity controller reads as

$$\boldsymbol{\tau}_{\text{fb}} = \mathbf{K}_{i,v}\int_0^t \left(\dot{\mathbf{e}}_q + \dot{\mathbf{q}}_c\right) \mathrm{d}\tilde{t} + \mathbf{K}_{p,v}\left(\dot{\mathbf{e}}_q + \dot{\mathbf{q}}_c\right) \; , \tag{7}$$

with the joint angle error

$$\mathbf{e}_q = \mathbf{q}_r - \mathbf{q} \tag{8}$$

and the positive definite controller gain diagonal matrices $\mathbf{K}_{p,v}$ and $\mathbf{K}_{i,v}$. The proportional position controller gives the output

$$\dot{\mathbf{q}}_c = \mathbf{K}_p(\mathbf{e}_q + \mathbf{u}) \; , \tag{9}$$

with the positive diagonal controller gain matrix $\mathbf{K}_p$. Moreover, $\mathbf{u}$ in (9) serves as input for the ILC scheme to be designed in the next section. The stability of computed-torque controllers is analyzed, e.g., in [23].

IV. ILC Algorithm

In this section, the online path ILC algorithm is discussed, and the implementation in discrete-time is presented.

Let $\boldsymbol{\gamma}(\lambda(t)) : [0;1] \mapsto \mathbb{R}^3$ denote a parametrized desired reference path in the task space of the robot. The user defines a desired execution speed by specifying the path parameter $\lambda(t) \in \mathcal{C}^2$. The tracking accuracy of the robot should be iteratively improved by the ILC each time the robot follows the trajectory $\boldsymbol{\gamma}(\lambda(t))$. Assume, without loss of generality, that the path parameter is normalized to the interval $[0;1]$ and the path is traversed in one direction, i.e., $\dot{\lambda} \geq 0$. Thus, the reference position of the TCP is given by $\mathbf{x}_r(t) = \boldsymbol{\gamma}(\lambda(t))$. The joint-space reference trajectory $\mathbf{q}_r(t) = \mathbf{f}^{-1}(\mathbf{x}_r(t))$ can be found by using the inverse kinematics.

The ILC law uses a proportional-derivative (PD) type update for iteration i at the time t of the form

$$\boldsymbol{\alpha}_i(t) = \mathbf{u}_{i-1}(t) + \mathbf{K}_{p,ilc}\mathbf{e}_{q,i}(\lambda(t)) + \mathbf{K}_{d,ilc}\dot{\mathbf{e}}_{q,i}(\lambda(t)) \; , \tag{10}$$

with the constant positive diagonal matrices $\mathbf{K}_{d,ilc}$ and $\mathbf{K}_{p,ilc}$. Details on the application, robustness, and convergence of this algorithm are given e.g., in [24], [25]. The joint velocity error $\dot{\mathbf{e}}_q$, see (8), can be obtained from the high precision encoders of the robot. Assuming small deviations from the path, the TCP position from (2) can be written by the Taylor expansion

$$\mathbf{x} = \mathbf{x}_r - \left.\frac{\partial \mathbf{f}(\mathbf{q})}{\partial \mathbf{q}}\right|_{\mathbf{q}=\mathbf{q}_r} \mathbf{e}_q + \mathcal{O}(\mathbf{e}_q)^2 \; . \tag{11}$$

Thus, from (11) and (3), the positioning error $\mathbf{e}_q$ can be calculated from the laser tracker measurement in the form

$$\mathbf{e}_q \approx \mathbf{J}_v^{-1}(\mathbf{q}_r)\mathbf{e} \; , \tag{12}$$

$$\mathbf{e} = \mathbf{x}_r - \mathbf{x} \; . \tag{13}$$

The ILC update in (10) is parametrized by the path parameter $\lambda(t)$. Since we assume $\dot{\lambda} \geq 0$, the path parameter $\lambda(t)$ is a monotonic function of t and can be used to calculate the angle correction values $\mathbf{u}_i$ of the ILC law for the corresponding position along the path. Remark: The assumption $\dot{\lambda} \geq 0$ can be alleviated by splitting the trajectory into parts with common traversing direction and considering them as separate repetitive control problems. Next, the ILC input from (10) is filtered with the Q-filter

$$\mathbf{u}_i = Q * \boldsymbol{\alpha}_i \; , \tag{14}$$

which filters out nonrepetitive effects and high-frequency measurement noise. The filter is realized by a Gaussian filter [26]

$$Q = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{\lambda^2}{2\sigma^2}\right) \; , \tag{15}$$

with the standard deviation

$$\sigma = \frac{\sqrt{\ln(2)}}{2\pi f} \, , \tag{16}$$

where the filter parameter f is the frequency at which the filter has a gain of -3dB.

A. Orientation Error of the Wrist Axes

Since the laser tracker cannot directly measure the orientation, the orientation of the end-effector is calculated from the motor encoder measurements. Hence, the axis-angle error of the wrist axes is

$$\mathbf{e}_{q,w} = \mathbf{q}_{w,r} - \mathbf{q}_w \; , \tag{17}$$

with the joint angles of the (wrist) axes $4, 5$, and 6 $\mathbf{q}_w = [q_4, q_5, q_6]^{\text{T}}$ and the corresponding reference $\mathbf{q}_{w,r}$. Analogously to (12), the wrist joint error $\mathbf{e}_{q,w}$ can be used in (10) to compensate for the orientation error. The proposed ILC concept can also be employed with a 6-dimensional measurement of the end-effector pose.

B. Online Implementation in Discrete-Time

In this section, the online implementation of the ILC law and the Q-filter is discussed. First, the parameter range $\lambda \in [0;1]$ is split into N equidistant intervals of the length $\delta = 1/N$. Let $\lambda_k = \lambda(kT_s)$ denote the path parameter $\lambda(t)$ at the time $t = kT_s$ with the sampling time T_s and $k \in \mathbb{Z}$. Further, $l_k = \text{round}(\lambda_k/\delta) \in [0; N-1]$ is the index value associated with the path parameter λ_k at the time kT_s. Subsequently, any variable x as a function of the discrete path parameter l

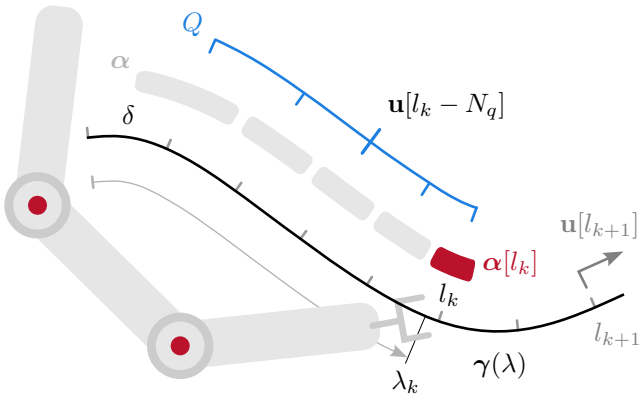

Fig. 3. Schematic of a path $\boldsymbol{\gamma}(\lambda)$, the current robot TCP position at the path parameter λ_k, the stored update data vector $\mathbf{u}$, and the filter Q.

is denoted by $x[l]$. The zero-order hold equivalent of (15) is given by

$$q_G[l] = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{l^2}{2\sigma^2}\right) . \tag{18}$$

This discrete-time filter is normalized over a square window with a total length of $2N_q + 1$ samples and applied to the ILC signal using the discrete-time convolution over $2N_q + 1$ path parameter indices, see also (14)

$$\mathbf{u}[k] = q * \boldsymbol{\alpha} = \sum_{m=-N_q}^{N_q} \frac{q_G[m]\boldsymbol{\alpha}[k-m]}{\sum_{m=-N_q}^{N_q} q_G[m]} . \tag{19}$$

The ILC algorithm is summarized in Algorithm 1, and a schematic of the algorithm is depicted in Fig. 3.

Algorithm 1 Online path ILC algorithm

1: $l_k \leftarrow$round(λ_k/δ)
2: **if** $l_k \neq l_{k-1}$ **then** ▷ new λ interval
3: $\quad \mathbf{e} \leftarrow \mathbf{x}_r - \mathbf{x}$
4: $\quad \boldsymbol{\alpha}[l_k] \leftarrow$(10) ▷ save $\boldsymbol{\alpha}$ at l_k
5: $\quad \mathbf{u}[l_k - N_q] \leftarrow$(19) ▷ filter at $l_k - N_q$
6: **end if**
7: $l_{k+1} \leftarrow$round$((\lambda_{k+1} + t_d)/\delta)$
8: **return** $\mathbf{u}[l_{k+1}]$ ▷ return ILC input

Due to the discretization of the path parameter, an interval of the path parameter can remain active for multiple time instances k. This is particularly clear when stopping the execution, i.e., $\dot{\lambda} = 0$ and $l_{k+1} = l_k$. Here, the algorithm is updated with the first encountered value for each interval. This is realized by updating each interval once per trial and otherwise outputting the same value repeatedly, see line 2 of Algorithm 1. In line 4, the currently measured error signal is mapped to the joint space of the robot and the ILC law (10) is applied. The calculated update is stored in the variable $\boldsymbol{\alpha}[l_k]$ at the current path parameter index l_k, see also Fig 3. Finally, the value at the path parameter index $l_k - N_q$ is filtered in line 5 which yields the ILC correction value $\mathbf{u}[l_k - N_q]$. The time delay t_d and the path parameter are used to calculate the next path parameter value $l_{k+1} = \text{round}((\lambda_{k+1} + t_d)/\delta)$. The

$\mathbf{K}_{p,v}$	= diag(1440, 1440, 500, 20, 14, 11)
$\mathbf{K}_{i,v}$	= diag(1200, 2560, 1700, 104, 75, 65)
$\mathbf{K}_p$	= 20 $\mathbf{I}$
$\mathbf{K}_{p,ilc}$	= 1 $\mathbf{I}$
$\mathbf{K}_{d,ilc}$	= 0.01 $\mathbf{I}$
N_q	= 45
f	= 5 Hz
T_s	= 1 ms
N	= 10000

TABLE I
CONTROL AND ILC PARAMETERS.

associated joint-angle corrections $\mathbf{u}[l_{k+1}]$ serve as the ILC input signal to compensate for the time delay of the system. The benefit of this recursive algorithm lies in the flexible calculation scheme and the computational efficiency.

V. EXPERIMENTAL VALIDATION

The 6 degrees-of-freedom industrial robot *Comau Racer 7-1.4* and a *Hexagon AT960* laser tracker were used for the experimental validation of the proposed online path ILC approach, see Fig. 1. It is well known that the joint-angle offsets play an essential role in the absolute accuracy of an industrial robot. In order to calibrate the joint positions, the robot was moved to a point on the reference trajectory, and a static calibration of all joints was performed by offsetting the axes to match the TCP position measured by the laser tracker. The orientation error is measured by the ZYX-Euler-angle representation $\mathbf{e}_o$ of the error between the reference and actual rotation matrix $\mathbf{R}_e = \mathbf{R}^e_{0,ref}(\mathbf{R}^e_0)^{\mathrm{T}}$. The control parameters for all experiments are summarized in Table I. Two experiments were conducted to validate the proposed ILC concept. In Experiment 1, the convergence and error caused by a variation in the execution speed of the robot is investigated. In Experiment 2, pre-recorded values are used for the feedforward control and the ILC law is applied without further usage of the laser tracker. In this experiment, the laser tracker measurement is used for validation only.

In Fig. 4, the reference path and the deviation from the path amplified by a factor of 50 are depicted for trial 0 and trial 1 of Experiment 1. The path covers a large part

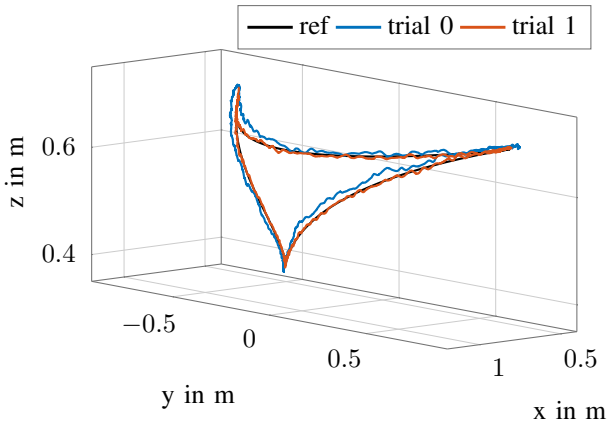

Fig. 4. Experiment 1: Robot trajectories with 50 times amplified deviation from the path. The reference path is drawn in black, trial 0 in blue, trial 1 in red.

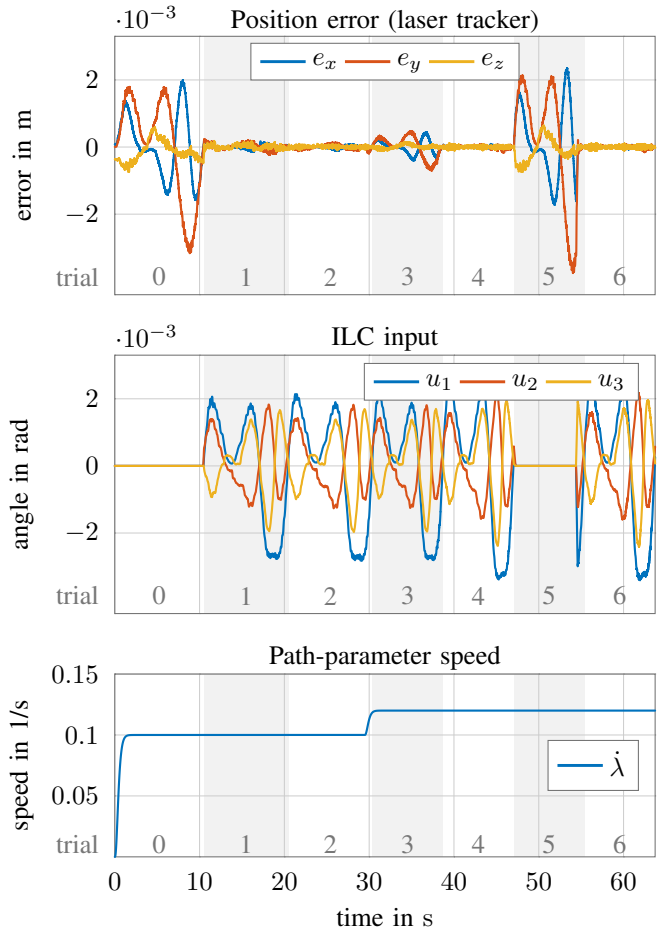

Fig. 5. Experiment 1 (ILC active): Position error, corresponding ILC input, and path-parameter speed for seven trials marked by the gray-shaded areas (trial 3: change in the robot's execution speed; trial 5: ILC turned off).

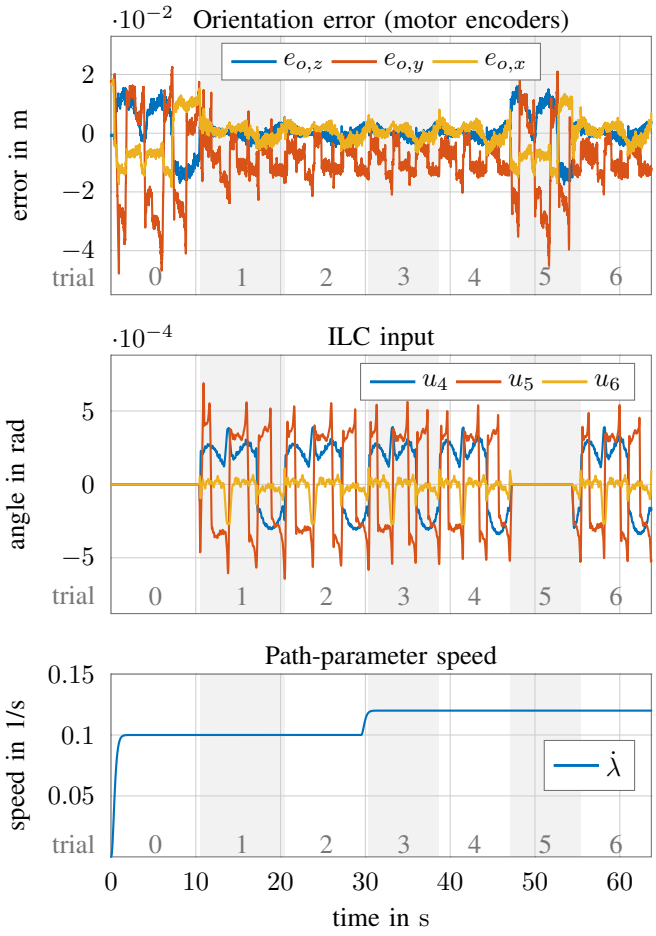

Fig. 6. Experiment 1 (ILC active): Orientation error, corresponding ILC input, and path-parameter speed for seven trials marked by the gray-shaded areas (trial 3: change in the robot's execution speed; trial 5: ILC turned off).

of the workspace and contains segments where $\frac{\partial \gamma}{\partial \lambda}$ is small, resulting in slow movements at a constant parameter speed. Conversely, the path also features fast movements where $\frac{\partial \gamma}{\partial \lambda}$ is large. There are three reversing points, one in the direction of the gravitational acceleration. These features render the reference trajectory challenging for a high-precision application, since friction effects and joint elasticities play an essential role in this scenario.

The position error and the corresponding ILC input of Experiment 1 are given in Fig. 5, where the gray-shaded areas mark the seven trials. Table II lists the maximum and RMS errors for each trial. In Experiment 1, the maximum error improves by 92% after one trial. After trial 2, the path-parameter speed is increased from $\dot{\lambda} = 0.1$ to 0.12 trials/s. Despite this change in the execution speed of the robot, the maximum error of trial 3 is 76% lower compared to trial 0. The slightly increased error due to the change in the execution speed is reduced to the level achieved earlier after learning at this speed for one trial. In trial 4, the maximum error is reduced by 94% compared to trial 0. Finally, in trial 5, the ILC correction output and learning are deactivated to show the baseline error at this execution speed and the improvement using the ILC signals again in trial 6, where the ILC is activated. The maximum error throughout trials 2, 4, and 6 is lower than 200 um, and the RMS error is lower than 80 um.

In Fig. 6, the orientation error of Experiment 1 is depicted. After the first learning trial, the error improves by more than a factor of 2. This significant improvement is retained throughout trial 3, where the execution speed is increased by 20%. However, the orientation error does not converge further after the first trial. This is a consequence of the joint-space orientation ILC Fig. 17 based on the motor encoder measurements.

Experiment 2 is conducted with pre-learned ILC signals in a feedforward sense. No further learning is done in this experiment, and the laser tracker was used only to validate the accuracy that can be achieved when using the pre-learned signals as feedforward input. Fig. 7 shows the position error and the pre-learned ILC feedforward input for the six trials of Experiment 2. During the trials 1, 2, 3, and 5, a maximum accuracy of below 200 um is achieved without updating the ILC data. The maximum position and orientation errors and RMS errors during the six trials of Experiment 2 are listed in Table III. From this table, it is clear that the absolute accuracy of the robot can be drastically improved with the

trial	$\|\mathbf{e}\|_\infty$ in mm	RMS($\mathbf{e}$) in mm	$\|\mathbf{e}_o\|_\infty$ in 10^{-3} deg	RMS($\mathbf{e}_o$) in 10^{-3} deg
0	3.238	1.8599	51.92	26.09
1	0.258	0.1191	23.93	10.91
2	0.166	0.0767	21.56	10.31
3	0.749	0.3923	21.15	10.12
4	0.186	0.0564	20.40	10.19
5	3.842	2.1570	46.88	24.37
6	0.183	0.0564	21.03	10.21

TABLE II

POSITION AND ORIENTATION ERROR FOR THE SEVEN TRIALS OF EXPERIMENT 1.

trial	$\|\mathbf{e}\|_\infty$ in mm	RMS($\mathbf{e}$) in mm	$\|\mathbf{e}_o\|_\infty$ in 10^{-3} deg	RMS($\mathbf{e}_o$) in 10^{-3} deg
0	3.226	1.8493	48.73	25.18
1	0.196	0.0776	19.23	9.95
2	0.192	0.0773	19.92	9.98
3	0.194	0.0782	20.22	10.04
4	0.751	0.3922	20.91	9.97
5	0.199	0.0775	20.01	10.03

TABLE III

POSITION AND ORIENTATION ERROR FOR THE SIX TRIALS OF EXPERIMENT 2.

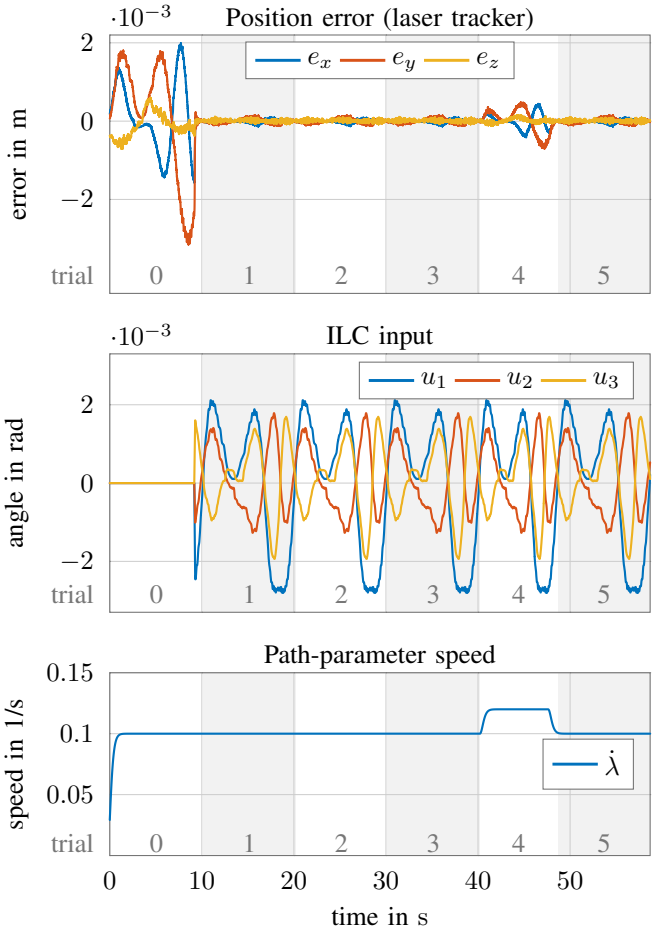

Fig. 7. Experiment 2 (ILC only used for feedforward control): Position error, corresponding pre-trained ILC feedforward input, and path-parameter speed for six trials marked by the gray-shaded areas (trial 4: change in the execution speed of the robot).

proposed method, even if an expensive laser tracker is only used for an initial learning phase.

In trial 4, the parameter speed is increased from $\dot{\lambda} = 0.1$ trials/s to 0.12 trials/s, but the feedforward signal was not changed to account for this higher execution speed. Here, only slightly larger errors occur. The accuracy is improved by more than a factor of 4 compared to trial 0, even with a 20% change in the execution speed of the robot. Finally, in trial 5, the execution speed is reduced again, and the execution speed matches the speed when the feedfoward signal was learned. Here, the same absolute accuracy is achieved, as in trials 1, 2, and 3.

VI. CONCLUSION

The contribution of this paper is the presentation and practical validation of a numerically efficient flexible online path ILC scheme. The key novelties are the online update of the PD-type ILC algorithm and the use of the path parameter as a parametrization of the ILC approach.

Using this approach, the practical experiments confirm a rapid convergence during continuous task execution with an accuracy improvement of 95% after two trials. The proposed path-parameter-based ILC can be easily employed for different and even varying execution speeds where a significant increase in execution accuracy is retained. Similarly, learning from partial trials is possible with the presented approach. Adaptating the ILC input to a different execution speed can be achieved with a single trial.

The experiments were conducted with an external laser tracker absolute measurement system. It is shown that the presented method can significantly improve the absolute accuracy of the robot. Due to the high repeatability of industrial robots the achieved accuracy can be sustained even without this measurement system and trial-by-trial learning. The orientation was measured using the motor encoders of the robot, highlighting the adaptation capabilities of the approach to different measurement scenarios. The maximum error of the orientation was reduced by a factor of 2. The presented approach shows multiple practical advantages and can be easily deployed to different path-tracking problems and robotic systems.

In summary, the presented approach shows multiple practical advantages and can be easily deployed to different path tracking problems and robotic systems. A drawback of the presented approach is the simple ILC-law. It is an open research question to combine a model-based learning filter with the presented path-ILC framework. Moreover, improving the achieved performance of the method for variations in execution speed is a topic to be investigated. In addition, applications with contacts between the manipulator and the environment present topics for future research. Further investigations could include the generalization of pre-learned ILC-data to different paths.

REFERENCES

[1] N. Schillreff, M. Nykolaychuk, and F. Ortmeier, "Towards High Accuracy Robot-Assisted Surgery," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 5666–5671, 2017.

[2] M. Garstenauer, C. Mittermayer, M. Reyhani-Masouleh, and M. Schwegel, "Shopfloor-Ready High Accuracy Robotics," in *International Symposium on Robotics Europe*, 2022, pp. 1–8.

[3] J. Na, Q. Chen, and X. Ren, "Chapter 6 - Adaptive Control for Manipulation Systems With Discontinuous Piecewise Parametric Friction Model," in *Adaptive Identification and Control of Uncertain Systems with Non-smooth Dynamics*. Massachusetts, USA: Academic Press, 2018, pp. 93–105.

[4] C. Hu, S. Lin, Z. Wang, and Y. Zhu, "Task Space Contouring Error Estimation and Precision Iterative Control of Robotic Manipulators," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7826–7833, 2022.

[5] L. Wang, T. Chai, and C. Yang, "Neural-Network-Based Contouring Control for Robotic Manipulators in Operational Space," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 4, pp. 1073–1080, 2012.

[6] F. Fujii, T. Nonomura, and T. Shiinoki, "Implementation of six degree-of-freedom high-precision robotic phantom on commercial industrial robotic manipulator," *Biomedical Physics & Engineering Express*, vol. 7, no. 5, pp. 1–8, 2021.

[7] J. Wu, B. Zhang, L. Wang, and G. Yu, "An iterative learning method for realizing accurate dynamic feedforward control of an industrial hybrid robot," *Science China Technological Sciences*, vol. 64, no. 6, pp. 1177–1188, 2021.

[8] R. Lee, L. Sun, Z. Wang, and M. Tomizuka, "Adaptive Iterative Learning Control of Robot Manipulators for Friction Compensation," *IFAC-PapersOnLine*, vol. 52, no. 15, pp. 175–180, 2019.

[9] Y.-H. Lee, S.-C. Hsu, T.-Y. Chi, Y.-Y. Du, J.-S. Hu, and T.-C. Tsao, "Industrial robot accurate trajectory generation by nested loop iterative learning control," *Mechatronics*, vol. 74, pp. 1–11, 2021.

[10] M. Hofer, L. Spannagl, and R. D'Andrea, "Iterative Learning Control for Fast and Accurate Position Tracking with an Articulated Soft Robotic Arm," in *IEEE International Conference on Intelligent Robots and Systems*, 2019, pp. 6602–6607.

[11] G. Sebastian, Z. Li, Y. Tan, and D. Oetomo, "On implementation of feedback-based PD-type iterative learning control for robotic manipulators with hard input constraints," in *IEEE International Conference on Control and Automation*, 2019, pp. 43–48.

[12] C. Liu, M. Wang, X. Li, and S. Ratchev, "Feedforward Enhancement through Iterative Learning Control for Robotic Manipulator," in *IEEE International Conference on Automation Science and Engineering*, 2021, pp. 1067–1072.

[13] G. Litak and M. I. Friswell, "Vibration in gear systems," *Chaos, Solitons & Fractals*, vol. 16, no. 5, pp. 795–800, 2003.

[14] O. Koç, G. Maeda, and J. Peters, "Optimizing the Execution of Dynamic Robot Movements With Learning Control," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 909–924, 2019.

[15] B. Nemec, T. Petrič, and A. Ude, "Force adaptation with recursive regression Iterative Learning Controller," in *IEEE International Conference on Intelligent Robots and Systems*, 2015, pp. 2835–2841.

[16] J. Xia, Y. Li, L. Yang, and D. Huang, "Spatial Repetitive Learning Control for Trajectory Learning in Human-Robot Collaboration," in *IEEE Conference on Decision and Control*, 2019, pp. 5568–5573.

[17] N. Tanimoto, M. Sekimoto, S. Kawamura, and H. Kimura, "Generation of feedforward torque by reuse of ILC torque for three-joint robot arm in gravity," in *Conference of the Society of Instrument and Control Engineers of Japan*, 2017, pp. 927–931.

[18] S. Kawamura and N. Fukao, "A time-scale interpolation for input torque patterns obtained through learning control on constrained robot motions," in *IEEE International Conference on Robotics and Automation*, 1995, pp. 2156–2161.

[19] F. Boeren, A. Bareja, T. Kok, and T. Oomen, "Frequency-Domain ILC Approach for Repeating and Varying Tasks: With Application to Semiconductor Bonding Equipment," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 6, pp. 2716–2727, 2016.

[20] J. Liu, X. Dong, D. Huang, and M. Yu, "Composite Energy Function-Based Spatial Iterative Learning Control in Motion Systems," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 5, pp. 1834–1841, 2018.

[21] F. H. Kong, A. M. Boudali, and I. R. Manchester, "Phase-indexed ILC for control of underactuated walking robots," in *IEEE Conference on Control Applications*, 2015, pp. 1467–1472.

[22] Y. Wang, F. Gao, and F. J. Doyle, "Survey on iterative learning control, repetitive control, and run-to-run control," *Journal of Process Control*, vol. 19, no. 10, pp. 1589–1600, 2009.

[23] R. Kelly, V. Santibáñez, and A. Loría, *Control of Robot Manipulators in Joint Space*. London, England: Springer, 2005.

[24] R. W. Longman, "Iterative learning control and repetitive control for engineering practice," *International Journal of Control*, vol. 73, no. 10, pp. 930–954, 2000.

[25] Z. Bien and J.-X. Xu, *Iterative Learning Control - Analysis, Design, Integration and Applications*. Massachusetts, USA: Springer, 1998.

[26] W. Messner, R. Horowitz, W.-W. Kao, and M. Boals, "A new adaptive learning rule," *IEEE Transactions on Automatic Control*, vol. 36, no. 2, pp. 188–197, 1991.