Loughborough University

This item was submitted to Loughborough's Research Repository by the author.
Items in Figshare are protected by copyright, with all rights reserved, unless otherwise indicated.

Feedforward enhancement through iterative learning control for robotic manipulator

PLEASE CITE THE PUBLISHED VERSION

https://doi.org/10.1109/CASE49439.2021.9551523

PUBLISHER

IEEE

VERSION

AM (Accepted Manuscript)

PUBLISHER STATEMENT

© 2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

LICENCE

All Rights Reserved

REPOSITORY RECORD

Liu, Chengyuan, Mingfeng Wang, Xuefang Li, and Svetan Ratchev. 2021. “Feedforward Enhancement Through Iterative Learning Control for Robotic Manipulator”. Loughborough University. https://hdl.handle.net/2134/17009078.v1.

Feedforward Enhancement through Iterative Learning Control for Robotic Manipulator

Chengyuan Liu[1], Mingfeng Wang[1,2], Xuefang Li[3*] and Svetan Ratchev[1]

***Abstract*— This work presents an iterative learning control (ILC) algorithm to enhance the feedforward control (FFC) for robotic manipulators. The proposed ILC algorithm enables the cooperation between the ILC, inverse dynamics, and a PD feedback control (FBC) module. The entire control scheme is elaborated to guarantee the control accuracy of the first implementation; to improve the control performance of the manipulator progressively with successive iterations; and to compensate both repetitive and non-repetitive disturbances, as well as various uncertainties. The convergence of the proposed ILC algorithm is analysed using a well established Lyapunov-like composite energy function (CEF). A trajectory tracking test is carried out by a seven-degree-of-freedom (7-DoF) robotic manipulator to demonstrate the effectiveness and efficiency of the proposed control scheme. By implementing the ILC algorithm, the maximum tracking error and its percentage respect to the motion range are improved from 5.78° to 1.09°, and 21.09% to 3.99%, respectively, within three iterations.**

I. INTRODUCTION

The involvement of internet of things (IoT) and internet of services (IoS) in the fourth industrial revolution (Industry 4.0) facilitates the development of the smart manufacturing, which has contributed to the widespread use of the robotic manipulators. At the same time, the smart manufacturing programme also puts forward new requirements and challenges on robotic manipulators, particularly in the area of precision control [1], [2]. Although the robotic manipulators provide extra flexibility and possibility of human-robot collaboration in manufacturing process, its absolute accuracy becomes a major factor that preventing its adoption in high precision industries, e.g. aerospace manufacturing. Improving the control accuracy of the robotic manipulators is challenging due to both internal and external uncertainties in the operation scenarios such as automated inspection, composite fabrication, additive manufacturing, etc. As a result, introducing advanced control scheme into manipulator controller becomes a generic and essential need for improving the robot manipulation precision.

The conventional controller of robotic manipulators has two streams: feedforward control (FFC) and feedback control (FBC), depending on whether the current measurement signal is used. The FFC provides computed joint input from the desired robot motion in real time, and the FBC gives joint input compensation by taking advantage of both the desired and the measured motion.

In the literature, due to the ability in handling system uncertainties and external disturbances, most of the existing works focus on the design and improvement of the FBC module in the control of robotic manipulators. For instance, many advanced FBC schemes, such as model predictive control [3], adaptive fuzzy control [4], and neural network based control [5], have been adopted to improve the control precision. However, the FBC approaches encounter difficulties due to disturbances and modelling errors (e.g. the hysteresis effects [6]), which motivates us to explore the ability of the FFC and thus improve the control performance of the whole system.

In robotic manipulators, the traditional FFC usually adopts the pure inverse dynamics, in which the control performance highly depends on the accuracy of the inverse dynamics. While in practice the accurate inverse dynamics is difficult to be obtained. To overcome this inherent disadvantage in the traditional FFC, various advanced control approaches have been investigated to compensate the inaccuracy in system model and thus to improve the performance of the FFC. For example, a fuzzy adaptation algorithm is developed in [7] to compensate the uncertainties existing in dynamics. In [8], the authors present an adaptive compensation method to handle the dynamic friction of the system. However, it should be noted that the above mentioned control methods generally require an intensive tuning process to improve the control performance. They also ignore the system repetition and do not possess the ability of learning from the past control experience. Inspired by these observations, several learning-based control approaches have been recently proposed, e.g. reinforcement learning [9] and neural networks [10], by considering the repetitive operation manners. These approaches have shown their superiority in handling the system uncertainties for robotics. While the main disadvantages of these methods lie in the time-consuming training process and the unavoidable nonsmoothness in control signals, which motivates the present work.

As an intelligent control approach, iterative learning control (ILC) has been proven to be effective to deal with high-precision control problems. By considering the system repetition, ILC is able to learn from the past control experience and thus improve the current control performance, despite the presence of various system uncertainties. From control theory point of view, ILC is actually a partial model-free control method, which may achieve a high-accuracy control

[1]Chengyuan Liu, Mingfeng Wang, and Svetan Ratchev are with the Institute for Advanced Manufacturing, University of Nottingham, Nottingham NG8 1BB, The U.K. chengyuan.liu@nottingham.ac.uk; svetan.ratchev@nottingham.a c.uk

[2]Mingfeng Wang is currently with the Department of Mechanical and Aerospace Engineering, Brunel University London, London UB8 3PN, the U.K. mingfeng.danny.wang@gmail.com

[3]Xuefang Li is with the School of Intelligent Systems Engineering, Sun Yat-Sen University, Guangzhou, China xuefangli@u.nus.edu

This work is supported in part by the Innovate UK project Wing LIFT under the Grant number 113162.

performance by just utilising the system input and output information without requiring the precise system dynamics. Owing to these impressive characteristics, ILC has been intensively implemented in various practical applications in robotics [11]–[13]. Moreover, with the learning ability, ILC is capable of compensating repetitive disturbances and suppressing fast hysteresis effects [14]. Recently, targeting at the FFC of industrial manipulators, several works have been reported to show the superiority of ILC. For instance, an ILC approach combined with a H_∞ FBC is proposed in [15] to improve the control performance of nanopositioning system. In [16], the authors establish an ILC scheme in combination with low-gain FBC to improve tracking accuracy for robotic manipulators. A norm-optimal ILC scheme for articulated soft manipulators is presented in [17] to improve the tracking control performance. It is worth to note that in most of the above mentioned works, the ILC schemes are designed directly to replace the role of the pure inverse dynamics in the traditional controllers, which however did not fully take the advantage of the available system dynamics. Motivated by this observation, it is of practical importance to develop new controllers which fully utilise the available system knowledge as well as the learning ability.

In the present work, a novel learning-based controller is developed for the tracking control of robotic manipulators. It consists of three parts including a FBC module, an inverse dynamics module, as well as an ILC strategy. In the proposed control approach, the ILC algorithm is used to enhance FFC performance achieved by the inverse dynamics through compensating the inaccuracies in the system model, while the FBC is adopted to nullify the effect of non-repeatable external disturbances. In contrast to existing works, the main contributions of this work can be summarised as follows:

1) A novel learning-based controller is developed to enable the coordinating work of the inverse dynamics, the FBC and the ILC strategy. By fully utilising the system knowledge and the system repetition, the proposed controller possesses the ability to achieve high-precision tracking tasks despite the presence of various system uncertainties and external disturbances.
2) In contrast to the existing works [15]–[19] where the ILC is utilised to replace the inverse dynamics directly, the proposed ILC scheme is designed to work in parallel with the inverse dynamics to fully use the available system information. Hence, it is able to enhance the control performance of the FFC by nullifying the effect of the system uncertainties.
3) The rigorous convergence analysis of the proposed control scheme is conducted by establishing a Lyapunov-like composite energy function (CEF). It has been shown explicitly that the cooperation between the ILC and FBC is capable of reducing the peak error and improving the tracking accuracy significantly as the iteration number increases.
4) The effectiveness of the proposed control scheme is verified through both simulations and experiments. In the later one, the proposed learning-based controller is implemented on a seven-degree-of-freedom (7-DoF) robotic manipulator. In contrast to the demonstrated results in [3], where only three joints of the manipulator are used, we consider all the seven joints of the demonstrated manipulator.

The rest of the paper is organised as follows. The robotic system, consisting of the kinematics and dynamics, is presented in Section II. In Section III, the control scheme is proposed with rigorous convergence analysis using a well established CEF. In Section IV, an experiment test on a 7-DoF robotic manipulator is carried out to show the effectiveness and efficiency of the proposed control scheme. Section V concludes the present work and future directions. Throughout the present work, let's denote $\mathbb{N}$ and $\mathbb{R}$ the set of natural and real numbers, respectively. For a given matrix H, H^T denotes the transformation of H, and $H \succ 0$ represents that H is positive definite.

II. System Description

A. System Overview

The robotic manipulator utilised in this work is the KUKA LBR IIWA 14 R820 robot (named as the IIWA robot in the rest of the work), as shown in Fig. 1, which is mounted on a mobile platform with both levelling feet and wheels. The IIWA robot consists of $n = 7$ revolute (R-) joints providing 7-DoF manipulation at the end-effector and in each R-joint, a torque sensor and an angular position encoder are equipped after the corresponding gearbox. The kinematic sketch is illustrated in Fig. 1 and the corresponding Denavit-Hartenberg (D-H) parameters are listed in TABLE I.

In this work, the CoppeliaSim® is selected for both simulation and real robot control purpose, due to its supporting of simultaneous incorporation with both programming environment (i.e. MATLAB® in this work) and hardware control.

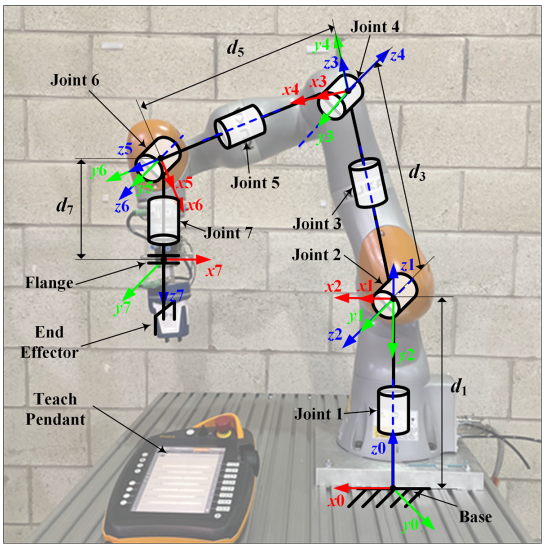


Fig. 1: Overview of the IIWA robot and the corresponding kinematic sketch with D-H coordinate system.

B. Dynamic Modelling

Consider the following n DoF dynamics of the robotic manipulator

$$M(q_k(t))\ddot{q}_k(t)+C(q_k(t),\dot{q}_k(t))\dot{q}_k(t)+G(q_k(t))=\tau_k(t), \quad (1)$$

where $t \in \mathbb{R}$ denotes the time instant and $k \in \mathbb{N}^+$ denotes the number of iteration. $q_k(t), \dot{q}_k(t), \ddot{q}_k(t) \in \mathbb{R}^n$ are the joint position, velocity and acceleration vectors, respectively, at time t of the kth iteration. $\tau_k(t)$ is the vector of joint torques, which is the system control input. $M(q_k(t)) \in \mathbb{R}^{n\times n}$ is the inertia matrix, which is symmetric, bounded and positive definite, $C(q_k(t),\dot{q}_k(t))\dot{q}_k(t) \in \mathbb{R}^n$ represents the Coriolis and centrifugal forces, $G(q_k(t)) \in \mathbb{R}^n$ represents the gravitational and frictional forces. For the dynamics (1), the following properties always hold:

- P1 $M(q) = M^T(q) \succ 0$ for any $q \in \mathbb{R}^n$, and there exist constants $\mu_1 > 0$ and $\mu_2 > 0$ such that $0 \preceq \mu_1 I_n \preceq M(q) \preceq \mu_2 I_n$, where I_n denotes the identity matrix of dimension n.
- P2 The matrix $(\frac{1}{2}\dot{M}(q_k(t)) - C(q_k(t),\dot{q}_k(t)))$ is skew symmetric and satisfying

$$x^T(\frac{1}{2}\dot{M}(q_k(t))-C(q_k(t),\dot{q}_k(t)))x=0, \forall x\in\mathbb{R}^n. \quad (2)$$

Assume that $q_d(t)$ is the desired trajectory, to facilitate the controller design and subsequent convergence analysis, the following two assumptions are imposed to the system:

- A1 $q_d(t)$ is first and second order continuously differentiable for all $t \in [0, T]$, and $q_d(t), \dot{q}_d(t), \ddot{q}_d(t)$ are bounded for all $t \in [0, T]$.
- A2 The initial conditions $q_k(0) = q_d(0)$ and $\dot{q}_k(0) = \dot{q}_d(0)$ are satisfied for all $k \in \mathbb{N}$.

III. Controller Design

In the present work, to improve the control performance of the robotic manipulator, a novel learning-based controller is proposed to realise the high-precision tracking task with the presence of system uncertainties and external disturbances. In the proposed control algorithm, as illustrated in Fig. 2, the torque input τ_k to the robotic manipulator consists of three components including the learning controller $\tau_k^{ILC}(t)$, the inverse-dynamics-based controller $\tau_k^{ID}(t)$ and the FBC $\tau_k^{FB}(t)$, namely,

$$\tau_k(t) = \tau_k^{ILC}(t) + \tau_k^{ID}(t) + \tau_k^{FB}(t), \quad (3)$$

TABLE I: D-H parameters of the IIWA robot

Joint i	$a_i(m)$	$\alpha_i(^\circ)$	$d_i(m)$	$\theta_i^*(^\circ)$
1	0	0	0.36	0 ∈ [-170, 170]
2	0	-90	0	20 ∈ [-120, 120]
3	0	90	0.42	0 ∈ [-170, 170]
4	0	90	0	-90 ∈ [-120, 120]
5	0	-90	0.40	0 ∈ [-170, 170]
6	0	-90	0	60 ∈ [-120, 120]
7	0	90	0.187	0 ∈ [-175, 175]

$*$ The initial setting of θ_i are demonstrated in Fig. 1 with its corresponding joint motion range.

with

$$\tau_k^{ID}(t)=M(q_d(t))\ddot{q}_d(t)+C(q_d(t),\dot{q}_d(t))\dot{q}_d(t)+G(q_d(t)), \quad (4)$$

$$\tau_k^{FB}(t)= K_p e_k(t) + K_d\dot{e}_k(t), \quad (5)$$

$$\tau_k^{ILC}(t)= \tau_{k-1}^{ILC}(t) + \gamma\tau_{k-1}^{FB}(t), \quad (6)$$

where $q_d(t)$ is the desired trajectory, $\dot{q}_d(t)$ and $\ddot{q}_d(t)$ represent the desired joint velocity and acceleration, $K_p \succ 0$ and $K_d \succ 0$ are the diagonal control gain matrices to be designed, $e_k(t) := q_d(t) - q_k(t)$ represents the tracking error at the kth iteration, $\gamma > 0 \in \mathbb{R}$ is the learning gain to be determined. Moreover, without loss of generality, the initial value of the the proposed learning controller is assumed to be $\tau_0^{ILC} = 0$.

Remark 1: Note that the inverse dynamics will give the same result with the same reference signal (i.e. $q_d(t)$). Hence, $\tau_k^{ID}(t)$ is identical for every k and thus will not affect the performance of the ILC and the FBC. While the coordination of the inverse dynamics and the PD controller can guarantee the control precision of the first iteration.

Remark 2: It is worth to note that usually when implementing the FBC and the ILC algorithm together, they would mutually influence each other since the accumulation of the noise caused by derivative action through the iterations, as shown in [20]. In the present work, to reduce the interaction effect, the ILC controller is designed in the structure of (6), where the ILC input is corrected by the FBC signal at the previous iteration.

With the proposed controller (3)-(6), we have the result as shown below.

Theorem 1: Consider the robotic manipulator (1) satisfy the A1 - A2, the control laws (3) - (6) ensure the convergence of the tracking error as the iteration number approaches to infinity, namely,

$$\lim_{k\to\infty} e_k(t) = 0, \quad \lim_{k\to\infty} \dot{e}_k(t) = 0, \quad (7)$$

hold for all $t \in [0, T]$.

In the following, the time variable t will be omitted for clarity if no confusion caused. To pave the way for convergence analysis, the following notations are firstly defined:

$$\bar{\tau}_d := M(q_k)M^{-1}(q_d)\tau_d, \quad (8)$$

$$\xi_k := \dot{e}_k + K_d^{-1}K_p e_k, \quad (9)$$

where τ_d is the desired control input. Obviously, according to P1 and A1 - A2, it is not difficult to obtain that

$$\lim_{k\to\infty} q_k = q_d \implies \lim_{k\to\infty} \bar{\tau}_d = \tau_d, \quad (10)$$

$$\lim_{k\to\infty} \xi_k = 0 \implies \lim_{k\to\infty} e_k = 0. \quad (11)$$

Therefore, we may focus on the convergence of ξ_k and then obtain the convergence of e_k. The convergence analysis is based on the Lyapunov-like theory with the well-designed CEF presented as follows

$$J_k(t) := \int_0^t e^{-\lambda\iota}(\tilde{\tau}_k^{ILC})^T(\tilde{\tau}_k^{ILC})\mathrm{d}\iota, \quad (12)$$

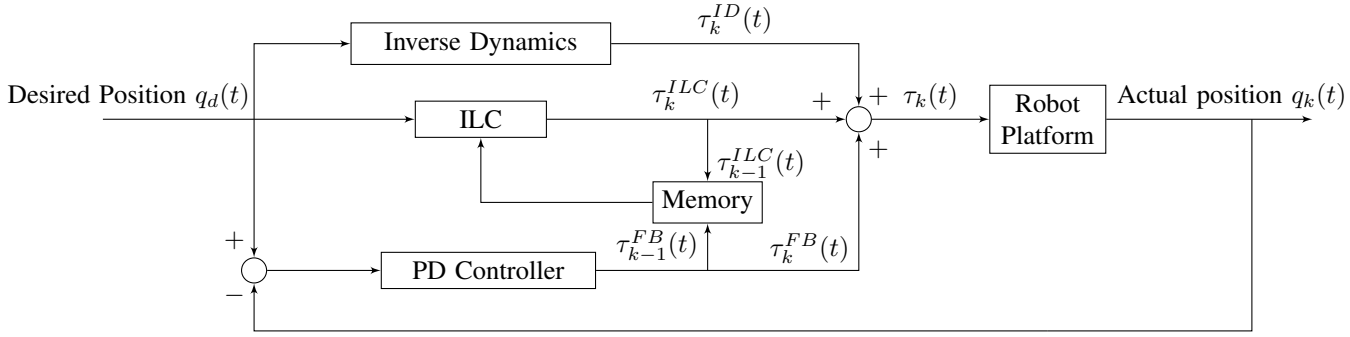


Fig. 2: The proposed control scheme consisting of the ILC, inverse dynamics and FBC.

where $\lambda \in \mathbb{R}$ is a positive constant and

$$\tilde{\tau}_k^{ILC} := \bar{\tau}_d - \tau_k^{ID} - \tau_k^{ILC}. \tag{13}$$

Proof of Theorem 1. The proof consists of two parts. The first part demonstrates the boundedness and non-increasing property of the $J_k(t)$ with respect to k. The second part shows the convergence of ξ_k and thus the convergence of the tracking errors.

Part I. From the definition in (5), (6) and (9), we have the difference of the CEF at a time instant $t \in [0, T]$ of two consequent iterations is

$$\begin{aligned}\Delta J_k = J_k - J_{k-1} &= \int_0^t e^{-\lambda\iota}(|\tilde{\tau}_k^{ILC}|^2 - |\tilde{\tau}_{k-1}^{ILC}|^2)\mathrm{d}\iota \\ &= \int_0^t e^{-\lambda\iota}(|\gamma K_d \xi_{k-1}|^2)\mathrm{d}\iota \\ &\quad - 2\int_0^t e^{-\lambda\iota}(\gamma K_d \xi_{k-1})^T(\tilde{\tau}_{k-1}^{ILC})\mathrm{d}\iota.\end{aligned} \tag{14}$$

Differentiating ξ_{k-1} followed by a multiplication with the inertia matrix $M(q_k)$, results in the following relationship:

$$\begin{aligned}M(q_{k-1})\dot{\xi}_{k-1} &= M(q_{k-1})\ddot{e}_{k-1} + M(q_{k-1})K_d^{-1}K_p\dot{e}_{k-1} \\ &= \tilde{\tau}_{k-1}^{ILC} - \tilde{\tau}_{k-1}^{FB} - C(q_{k-1}, \dot{q}_{k-1})\xi_{k-1} + \eta_{k-1},\end{aligned} \tag{15}$$

where

$$\tilde{\tau}_k := \bar{\tau}_d - \tau_k, \tag{16}$$

$$f(q_k, \dot{q}_k) := -M^{-1}(q_k)(C(q_k, \dot{q}_k)\dot{q}_k + G(q_k)) \tag{17}$$

$$\tilde{f}(q_k, \dot{q}_k) := f(q_d, \dot{q}_d) - f(q_k, \dot{q}_k), \tag{18}$$

and

$$\begin{aligned}\eta_{k-1} =& C(q_{k-1}, \dot{q}_{k-1})\xi_{k-1} + M(q_{k-1})\tilde{f}(q_{k-1}, \dot{q}_{k-1}) \\ &+ M(q_{k-1})K_d^{-1}K_p\dot{e}_{k-1}.\end{aligned} \tag{19}$$

Substitute (15) into (14), following several manipulations, we can obtain that

$$\begin{aligned}\Delta J_k =& -e^{-\lambda t}(\gamma K_d \xi_{k-1})^T(M(q_{k-1})\xi_{k-1}) \\ &- \lambda\int_0^t e^{-\lambda\iota}(\gamma K_d \xi_{k-1})^T(M(q_{k-1})\xi_{k-1})\mathrm{d}\iota \\ &- 2\int_0^t e^{-\lambda\iota}(\gamma K_d \xi_{k-1})^T(K_d \xi_{k-1})\mathrm{d}\iota \\ &+ \int_0^t e^{-\lambda\iota}(\gamma K_d \xi_{k-1})^T(\gamma K_d \xi_{k-1} + 2\eta_{k-1})\mathrm{d}\iota.\end{aligned} \tag{20}$$

From the definition (9) and (19), it is possible to show that there exist two constants $c_1 > 0$ and $c_2 > 0$ [21] such that

$$|\eta_{k-1}| \le c_1|\xi_{k-1}| + c_2|\xi_{k-1}|^2. \tag{21}$$

The property P1 shows that there exists a constant $\mu > 0 \in \mathbb{R}$ such that for any bounded q_{k-1},

$$0 \preceq \mu I_n \preceq K_d^T M(q_{k-1}). \tag{22}$$

Using the inequalities (21) and (22) for (20) gives

$$\begin{aligned}\Delta J_k \le& -\mu\gamma e^{-\lambda t}|\xi_{k-1}|^2 - (\lambda+2)\gamma\int_0^t e^{-\lambda\iota}|\xi_{k-1}|^2\mathrm{d}\iota \\ &+ \gamma\int_0^t e^{-\lambda\iota}h(|\xi_{k-1}|)|\xi_{k-1}|^2\mathrm{d}\iota,\end{aligned} \tag{23}$$

where $h(|\xi_{k-1}|) = |K_d|(\gamma|K_d| + 2c_1 + 2c_2|\xi_{k-1}|)$. Since there exists a positive constant $c_3 > 0$ such that $0 < h(|\xi_{k-1}|) < c_3$, we have

$$\Delta J_k \le -(\lambda + 2 - c_3)\gamma\int_0^t e^{-\lambda\iota}|\xi_{k-1}|^2\mathrm{d}\iota - \mu\gamma e^{-\lambda t}|\xi_{k-1}|^2. \tag{24}$$

It follows that if λ is selected such that $\lambda + 2 - c_3 > 0$, then

$$\Delta J_k \le -\mu\gamma e^{-\lambda t}|\xi_{k-1}|^2 \le 0, \tag{25}$$

which shows the non-increasing property of the CEF J_k.

Part II. Computing the cumulative sum for (25) gives that

$$\sum_{i=1}^{k}\Delta J_i(t) \le -\mu\gamma\sum_{i=1}^{k} e^{-\lambda t}|\xi_{i-1}(t)|^2, \tag{26}$$

which implies

$$J_k(t) \le J_{-1}(t) - \mu\gamma\sum_{i=1}^{k} e^{-\lambda t}|\xi_{i-1}(t)|^2, \tag{27}$$

Since J_{-1} is bounded and J_k is positive, we have for $t \in [0, T]$,

$$\lim_{k\to\infty}|\xi_{i-1}(t)| = 0, \tag{28}$$

which implies the convergence (7).

IV. Experimental Validation

A. Trajectory and Data Acquisition

To validate the proposed control scheme in Section III , a small scale ellipse trajectory in the XY-plane is selected for tracking control, as it requires synchronised actuation of all the 7 joints. The ellipse trajectory contains 9665 samples in total and the duration is $9.8\ s$. Inverse kinematics of the IIWA robot is utilised to compute the desired joint position, then a low pass filter is used to generate the corresponding joint velocity and acceleration. The joint position, velocity and input torque range responding to the ellipse path are listed in TABLE II.

To implement the proposed control algorithm, the MATLAB® is deployed to compute the torque input, which is subsequently delivered to the virtual robot in CoppeliaSim®. Once the simulation results convince of the safety and effectiveness of the controller, the control signal can be then transferred to the real IIWA robot through the UDP socket. Meanwhile, the joint positions of the IIWA robot are acquired through its integrated joint sensors.

In the purpose of demonstrating the effectiveness of the ILC algorithm, comparison in the trajectory tracking between the initial implementation (Case I) and the third iteration (Case II) are considered. Note that at the initial implementation, only inverse dynamics and the FBC are involved in the system, i.e. $\tau_0^{ILC} = 0$. The repetitive disturbances, i.e. the kinematic uncertainty and initial joint offset, are considered during the experimental test. Furthermore, the environmental disturbance, i.e. platform with wheel configuration, is considered as non-repetitive disturbance.

B. Test results

Based on the aforementioned setup, the trajectory tracking tests are conducted in both simulation and real experiment. The joint positions measured during the the experimental test are presented in Fig. 3, where the black solid line shows the desired joint positions along time t, the blue dotted line represents the joint positions under the control of the inverse dynamics and the FBC (i.e. Case I), and the red dashed line shows the joint performance after three iterations with the ILC algorithm (i.e. Case II). The corresponding joint error value and its percentage respect to the motion range are listed in TABLE III in the form of average (minimum to maximum).

TABLE II: Joint position, velocity and torque ranges

Joint	$q(^\circ)$	$\dot{q}(^\circ/s)$	$\tau(Nm)$
1	[0.00, 9.42]	[-4.49, 5.23]	[-0.16, 0.31]
2	[6.91, 34.30]	[-10.44, 12.30]	[-37.17, -19.40]
3	[-2.21, 2.50]	[-2.40, 0.76]	[-0.36, 0.13]
4	[-106.36, -68.91]	[-14.52, 17.53]	[13.12, 14.09]
5	[-0.11, 0.75]	[-0.22, 0.42]	[-0.32, -0.22]
6	[71.70, 81.77]	[-4.49, 5.23]	[0.00, 0.06]
7	[-0.61, 11.74]	[-5.60, 3.88]	[-0.01, 0.01]

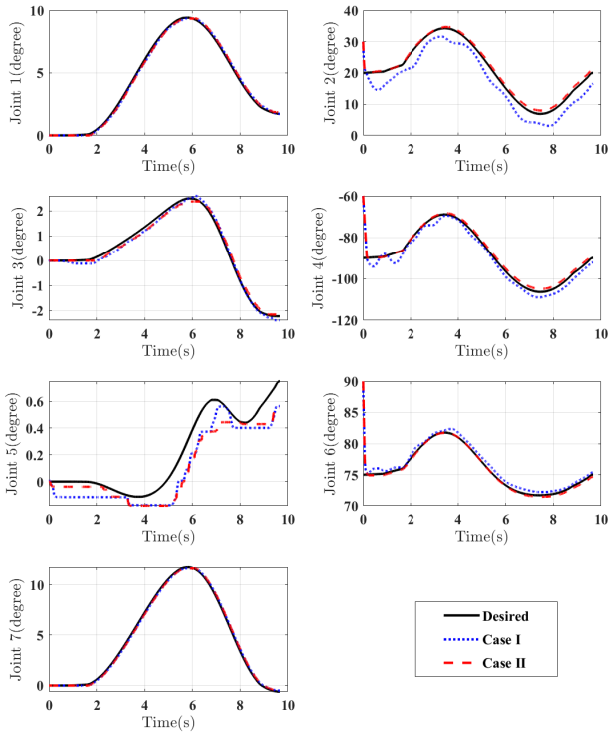


Fig. 3: Joint tracking trajectories represented by the black solid line for the desired, the blue dotted line for the Case I, and the red dashed line for the Case II.

As shown in Fig. 3, it can be noted that the initial offsets of joints 2, 4, and 6 have been set as 10°, 30° and 15°, respectively, which were utilised as system repetitive uncertainty. Moreover, as mentioned in Section II, the mobile platform for the IIWA robot equips with both levelling feet and wheels, and the later option is chosen for providing the nonrepetitive disturbance. As the consequence of the uncertainty and disturbance, oscillations are obviously observed in Case I, which results in a maximum tracking error of 5.78° in joint 2 and 5.14° in joint 4, respectively. While in Case II, the maximum tracking errors are reduced to 1.09° and 1.53° in joints 2 and 4, respectively.

Furthermore, it is worthy noting that in TABLE III, the maximum tracking error percentage in joint 2 is 21.09%, this may caused by the unwell designed PD gain parameters. While the involvement of the ILC algorithm provides efficient compensation in the control input, which improves the maximum tracking error percentage to 3.99% in only three iterations.

In the case of small joint motion range, e.g. joints 3 and 5, although the ILC algorithm does not provide impressive improvement in the control performance, there is no chattering phenomenon observed neither. This indicates that the proposed structure of the ILC algorithm can work coordinately with the FBC as discussed in Remark 2.

TABLE III: Joint tracking errors in terms of value and percentage respect to motion range in the form of avg (min, max)

Joint	Range (°)	Case I		Case II	
		Value (°)	Percentage (%)	Value (°)	Percentage (%)
1	9.42	0.13 (0.00, 0.29)	1.36 (0.00, 3.06)	0.14 (0.01, 0.26)	1.44 (0.05, 2.77)
2	27.39	3.57 (1.45, **5.78**)	13.02 (5.29, **21.09**)	0.66 (0.00, **1.09**)	2.39 (0.00, **3.99**)
3	4.71	0.11 (0.00, 0.21)	2.39 (0.00, 4.50)	0.11 (0.00, 0.22)	2.27 (0.03, 4.75)
4	37.45	2.05 (0.00, 5.14)	5.47 (0.00, 13.73)	0.90 (0.00, 1.53)	2.39 (0.00, 4.09)
5	0.86	0.12 (0.00, 0.25)	13.45 (0.00, 28.78)	0.11 (0.00, 0.25)	12.18 (0.52, 28.82)
6	10.08	0.53 (0.00, 0.93)	5.22 (0.00, 9.27)	0.16 (0.00, 0.37)	1.62 (0.00, 3.68)
7	12.35	0.14 (0.00, 0.30)	1.17 (0.02, 2.40)	0.15 (0.00, 0.31)	1.24 (0.03, 2.51)
Average	-	**0.95** (0.21, 1.84)	**6.01** (0.76 , 11.83)	**0.32** (0.00, 0.58)	**3.37** (0.09, 7.23)

V. CONCLUSIONS

In this work, an ILC algorithm is proposed to enhance the FFC performance for the robotic manipulators by working coordinately with the inverse dynamics module and the FBC module. Through this control scheme, both repetitive and non-repetitive disturbances, as well as various uncertainties can be suppressed. Meanwhile, both the FBC and ILC algorithm contribute to the system convergence as presented in the Lyapunov-like analysis.

To validate the proposed ILC algorithm, an ellipse trajectory tracking experiment has been implemented by a 7-DoF IIWA robot, in which all the 7 joints are actuated synchronously. The resulting maximum tracking error and percentage to the range have been improved from $5.78°$ to $1.09°$, and 21.09% to 3.99%, respectively, in only three iterations. It is indicated that the proposed ILC algorithm is capable of compensating the environmental disturbances and system uncertainties, enhancing the system control performance, and working coordinately with the FBC module. Therefore, through learning from the previous system behaviour, the proposed control scheme can improve the control accuracy with successive iterations and finally achieve high-precision tracking performance.

To further explore the effectiveness and efficiency of the proposed learning based control scheme, more complex experiment scenarios and advanced FBC algorithm will be considered in the future work. The system could also be extended in the presence of dynamic payload and physical constraints.

REFERENCES

[1] D. A. Bristow and A. G. Alleyne, "A high precision motion control system with application to microscale robotic deposition," *IEEE Transactions on Control Systems Technology*, vol. 14, no. 6, pp. 1008–1020, 2006.

[2] S. Devasia, E. Eleftheriou, and S. R. Moheimani, "A survey of control issues in nanopositioning," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 5, pp. 802–823, 2007.

[3] T. Faulwasser, T. Weber, P. Zometa, and R. Findeisen, "Implementation of nonlinear model predictive path-following control for an industrial robot," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 4, pp. 1505–1511, 2016.

[4] B. K. Yoo and W. C. Ham, "Adaptive control of robot manipulator using fuzzy compensator," *IEEE Transactions on fuzzy systems*, vol. 8, no. 2, pp. 186–199, 2000.

[5] T. Sun, H. Pei, Y. Pan, H. Zhou, and C. Zhang, "Neural network-based sliding mode adaptive control for robot manipulators," *Neurocomputing*, vol. 74, no. 14-15, pp. 2377–2384, 2011.

[6] M. Goldfarb and N. Celanovic, "Modeling piezoelectric stack actuators for control of micromanipulation," *IEEE Control Systems Magazine*, vol. 17, no. 3, pp. 69–79, 1997.

[7] Y. Fan, K. Xing, and X. Jiang, "Fuzzy adaptation algorithms' control for robot manipulators with uncertainty modelling errors," *Complexity*, 2018.

[8] A. Amthor, S. Zschaeck, and C. Ament, "High precision position control using an adaptive friction compensation approach," *IEEE Transactions on automatic control*, vol. 55, no. 1, pp. 274–278, 2009.

[9] Y. Ansari, M. Manti, E. Falotico, M. Cianchetti, and C. Laschi, "Multiobjective optimization for stiffness and position control in a soft robot arm module," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 108–115, 2017.

[10] D. Braganza, D. M. Dawson, I. D. Walker, and N. Nath, "A neural network controller for continuum robots," *IEEE transactions on robotics*, vol. 23, no. 6, pp. 1270–1277, 2007.

[11] M. Norrlof, "An adaptive iterative learning control algorithm with experiments on an industrial robot," *IEEE Transactions on robotics and automation*, vol. 18, no. 2, pp. 245–251, 2002.

[12] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE control systems magazine*, vol. 26, no. 3, pp. 96–114, 2006.

[13] Z. Bien and J.-X. Xu, *Iterative learning control: analysis, design, integration and applications*. Springer Science & Business Media, 2012.

[14] S. Yamada and H. Fujimoto, "Precise joint torque control method for two-inertia system with backlash using load-side encoder," *IEEJ Journal of Industry Applications*, vol. 8, no. 1, pp. 75–83, 2019.

[15] B. Helfrich, C. Lee, D. Bristow, X. Xiao, J. Dong, A. G. Alleyne, S. M. Salapaka, and P. M. Ferreira, "Combined h_{∞}-feedback and eterative learning control design with application to nanopositioning systems," in *2008 American Control Conference*. IEEE, 2008, pp. 3893–3900.

[16] F. Angelini, C. Della Santina, M. Garabini, M. Bianchi, G. M. Gasparri, G. Grioli, M. G. Catalano, and A. Bicchi, "Decentralized trajectory tracking control for soft robots interacting with the environment," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 924–935, 2018.

[17] M. Hofer, L. Spannagl, and R. D'Andrea. (2019) Iterative learning control for fast and accurate position tracking with an articulated soft robotic arm. [Online]. Available: https://arxiv.org/abs/1901.10187

[18] T. Meng and W. He, "Iterative learning control of a robotic arm experiment platform with input constraint," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 1, pp. 664–672, 2017.

[19] K. Yovchev, K. Delchev, and E. Krastev, "State space constrained iterative learning control for robotic manipulators," *Asian Journal of Control*, vol. 20, no. 3, pp. 1145–1150, 2018.

[20] T. A, "Adaptive iterative learning control for robot manipulators," *Automatica*, vol. 40, no. 7, pp. 1195–1203, 2004.

[21] 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 *2019 IEEE 15th International Conference on Control and Automation (ICCA)*. IEEE, 2019, pp. 43–48.