The final publication is available at Springer via https://doi.org/10.1007/s10846-017-0653-z

Data-Inferred Personalized Human-Robot Models for Iterative Collaborative Output Tracking

Jonathan Realmuto · Rahul B. Warrier · Santosh Devasia

Received: 10 February 2017 / Accepted: 7 June 2017

Abstract This article studies collaborative human- robot output tracking when the desired output is only known to the human but not to the robot controller. The main contribution of this article is to propose and establish convergence conditions for an iterative learning algorithm that updates the robot input using (i) the effect of the human action on the combined human-robot output tracking (which includes the effect of the human-response dynamics) and (ii) data-inferred human-robot models. This allows the iterative learning control (ILC) to be personalized for each individual human operator. Additionally, experimental results are presented to illustrate the iterative learning approach. Results show that, with the proposed approach, the robot can learn to collaboratively track the output with 10.0% error, which is close to twice the robot noise of 4.6% of the desired output. Furthermore, the data-inferred models provided evidence of the effect of the human operator's dynamics on the co-tracking task.

1 Introduction

The idea of robots working alongside humans is at the forefront of robotics research today. For example, robots are being developed to: (i) work with humans in industrial assembly lines [1], (ii) assist in performing daily tasks [2–6], (iii) aid in post-stroke rehabilitation [7], and (iv) jointly solve problems for search and rescue operations, construction, and space exploration, e.g., [8].

Supervised learning techniques, such as reinforcement learning (RL) for human-robot collaborative learning, imitate expert user demonstrations [9]. Such techniques usually require extensive training sessions for the human(s) to gain expertise in operating the robot, which may not be feasible in all collaborative learning applications. When the human is not an expert but aims to convey intent, the human action's effect on the combined human-robot output tracking includes the effect of the human-response dynamics [10]. Therefore, effective collaboration between humans and robots in these applications relies on correcting for the effects of the human-response dynamics to infer the intent, i.e., the desired output. The main contribution of the current work is to develop a novel iterative learning control (ILC) algorithm for collaborative human-robot output tracking when the human intent is not directly available to the robot controller.

Iterative learning control (ILC) [11,12] is readily applicable to collaborative learning of demonstrated trajectories, and has been well studied for programming by demonstration (PbD), for example in automated surgical tasks [13], in self-learning automobile cruise control [14], and in model-based intent-estimation for human-in-the-loop skill transfer [15,16]. Often in such cases, the human is considered an expert and the trajectory achieved by the human can be considered as a trajectory to be followed by the robot, e.g., [17]. In contrast, this article considers the use of ILC when the desired output is not known to the robot controller but

Jonathan Realmuto, Rahul B. Warrier, Santosh Devasia
Department of Mechanical Engineering, University of Washington, Seattle, WA USA - 98195

Jonathan D. Realmuto
E-mail: realmuto@uw.edu

Rahul B. Warrier
E-mail: warrierr@uw.edu

Santosh Devasia
E-mail: sdevasia@u.washington.edu

needs to be inferred iteratively based on the combined human-robot output tracking, which is affected by the human-response dynamics.

ILC methods originated in robotics applications, starting with early works, e.g., [12], along with the use of an inverse model of the robot dynamics in early works [18]. Such approaches converge to the desired output if the modeling error is small. Improvements of the model through parameter adaptation with data acquired during the iteration was studied in [19] for robotics application using a discrete time implementation. Such estimation of the model data can enable faster convergence for new trajectories as shown in [20]. In these discrete-time approaches, the model is inverted and the convergence depends on the system being minimum-phase (i.e., no zeros on the right hand side of the complex plane) to ensure that the inverse is stable, e.g., [21, 20]. More recent model-inversion-based iterative learning control use the noncausal inverse in the iteration law for nonminimum-phase systems [22]. In the frequency domain such noncausal inverses can be computed in the Fourier domain [23] as

$$u_{k+1}(\omega) \;=\; u_k(\omega) + \rho(\omega) G_r^{-1}(\omega) e_k(\omega), \tag{1}$$

where the robot input u is updated based on the output tracking error $e = q_d - q_r$, ρ is the iteration gain, q_d is the desired output and q_r is the achieved robot output. For example, convergence to the desired output can be guaranteed if the phase uncertainty in the model is less than 90 degrees and the iteration gain is sufficiently small [23]. Similar conditions on the phase were established using the discrete-Fourier formulation in [24]. In the frequency domain, accuracy of the robot models G_r can be increased by using system input-output information from the previous iteration step k to form data-based models,

$$G_r(\omega) = y_k(\omega)/u_k(\omega) \tag{2}$$

as shown in [25]. However, this input-output-based modeling is not suitable for human-robot collaboration (e.g., as shown in Fig. 1) since both the human and the robot models are needed, and when the desired output trajectory q_d that serves as an input to the human is not available to the robot controller, which limits the ability to estimate the human-response model G_H in Fig. 1.

Nominal models of the human-robot system using training data (where the desired output is known) could be used, or learned from previous demonstrations [15, 16]. Since the models of the human-response models can vary for each individual operator, the current work proposes a new personalized approach, using the effect of the human action on the achieved human-robot output trajectory tracking, to infer and correct for the human-robot dynamics during ILC for collaborative output tracking. This allows the ILC to be personalized for each individual human operator. Such correction of the human-response dynamics is important when the human operator is not an expert and cannot achieve perfect tracking without assistance. Note that with an expert operator, the demonstrated trajectory q can be close to the desired output q_d, and the robot can then aim to track the demonstrated trajectory q. In contrast, the proposed approach is applicable even when achieved tracking of q_d is poor, e.g., when the human operator is a novice. The proposed approach aims to model and iteratively correct for the human-response dynamics. Moreover, the article develops conditions for convergence of the proposed collaborative output tracking approach. Finally, the proposed approach allows varying level of assistance from the robot during the collaborative output tracking [26]. Such differing levels of assistance can be important, e.g., in rehabilitation robotics using active orthoses [27,28] and in robotic prostheses [29–32].

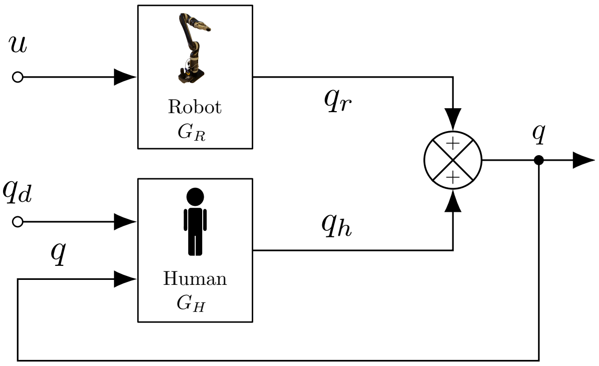

Fig. 1 Schematic diagram of collaborative output tracking.

The article is organized as follows. Section 2 formalizes the problem, presents the proposed iterative model inversion algorithm and develops convergence conditions for the proposed approach. The collaborative human-robot system, used to evaluate the proposed approach, is described in Section 3 along with implementation details. Experimental results are discussed in Section 4 and conclusions are in Section 5.

2 Problem formulation and solution

2.1 Problem setup

The objective is to learn the robot controller input u that achieves the robot output q_r required to track the desired output q_d, i.e.,

$$q(\cdot) \;=\; q_d(\cdot), \tag{3}$$

when the desired output q_d is known to the human but not known to the robot controller. Here, the total output q is a combination of the human output q_h and the robot output q_r (in the frequency domain)

$$q(\omega) \;=\; q_r(\omega) + q_h(\omega) \tag{4}$$

as illustrated in Fig. 1. The robot input-output relationship is modeled as

$$q_r(\omega) \;=\; G_R(\omega)u(\omega), \tag{5}$$

where G_R is the robot transfer function, and u is the input to the robot. Similarly, the difference between the desired output q_d and the achieved output q affects the human response, which is modeled as

$$q_h(\omega) \;=\; G_H(\omega)\left(q_d(\omega) - q(\omega)\right), \tag{6}$$

where G_H represents the human-response dynamics [10].

Note, that in (6), the input to the human operator is the (output) tracking error, e defined as,

$$e(\cdot) \;=\; q_d(\cdot) - q(\cdot). \tag{7}$$

In general, the human response dynamics may include multiple input channels including the desired trajectory, q_d (feed-forward loop) and the output, q (internal loop), but the response dynamics model based on the tracking error, as in (6), has been shown to be a reasonable assumption for low frequencies [33].

In terms of the tracking error, $e(\cdot)$, the objective of the collaborative task is to choose a suitable robot input u that in collaboration with the human input q_h achieves zero tracking error, $e(\cdot) = 0$. However, the research challenge is that the desired output q_d and therefore the tracking error e are only known to the human but not to the robot controller. The controller needs to infer the tracking error e (based on human action) and then reduce it.

2.2 Proposed Model-inversion-based ILC

The iterative learning control (ILC) approach updates the robot control input u (in the frequency domain for every iteration step $k \geq 0$) as,

$$u_{k+1}(\omega) \;=\; u_k(\omega) + \rho_k(\omega)\hat{G}_k^{-1}(\omega)e_{c,k}(\omega), \tag{8}$$

where ρ_k is the learning gain, $\hat{G}_k$ is the available model of the system, and $e_{c,k}$ is the estimate of the correction needed. If the desired trajectory q_d were known to the robot controller, then the correction needed $e_{c,k}$ could be chosen as the robot (desired-output) tracking error $e_{d,k}$, as

$$e_{d,k}(\omega) \;=\; \beta(\omega)q_d(\omega) - q_{r,k}(\omega) \tag{9}$$

with $\beta \in [0, 1]$ representing the level of assistance provided by the robot controller to track the desired output q_d, and is presumed to be pre-defined in this work. However, since the desired output q_d is not available to the robot controller, the correction term, $e_{c,k}$ in (8) is based on the difference between the total output q_k of the system and the contribution $q_{r,k}$ of the robot towards this output, i.e., the error term $e_{r,k}$

$$e_{r,k}(\omega) \;=\; \beta(\omega)q_k(\omega) - q_{r,k}(\omega). \tag{10}$$

Thus, with the error term $e_{r,k}$, defined in (10) as the correction term $e_{c,k}$, the collaborative update law (8) is modified to,

$$u_{k+1}(\omega) = u_k + \rho_k(\omega)\hat{G}_k^{-1}(\omega)\left[\beta(\omega)q_k(\omega) - q_{r,k}(\omega)\right]. \tag{11}$$

The research question is stated formally below.

Problem Statement: *Find the inverse model $\hat{G}_k^{-1}(\omega)$ during collaborative human-robot output tracking that ensures convergence of the ILC update law in* (11)*, when the desired output trajectory, q_d is not available to the robot controller.*

2.3 Impact of human-response dynamics on tracking

The human ability to track a trajectory influences the robot (desired-output tracking) error $e_{d,k}$ in (9) since the robot learns about the desired output trajectory q_d through human action. As shown below, the impact of the human-response dynamics G_H on the output tracking error $e_{d,k}$ reduces if the human is good at tracking the desired output or if the assistance level β of the robot is high (even if the human tracking ability is poor).

If the ILC with the update law (11) converges and the inverse system $\hat{G}_k^{-1}(\omega)$ and iteration gain $\rho_k(\omega)$ are nonzero at frequency ω, then the error term $e_{r,k}$ converges to zero, i.e.,

$$\lim_{k\to\infty} e_{r,k}(\omega) \;=\; 0, \tag{12}$$

with the robot providing the desired level of assistance β for the final output q^*, i.e.,

$$q_r^*(\omega) \;=\; \beta q^*(\omega) \tag{13}$$

where

$$q_r^*(\omega) \;=\; \lim_{k\to\infty} q_{r,k}(\omega), \quad q^*(\omega) \;=\; \lim_{k\to\infty} q_k(\omega). \tag{14}$$

Note that a relationship between total output q_k in (4) and the error term $e_{r,k}$ can be established through the input-output relationship for the control scheme in Fig. 1 as

$$q_k(\omega) \;=\; \left(\frac{1}{1+G_H(\omega)}\right) q_{r,k}(\omega) + \left(\frac{G_H(\omega)}{1+G_H(\omega)}\right) q_d(\omega). \tag{15}$$

Using (13) and (14), the converged solutions satisfy, by taking limits $k \to \infty$ on both sides of (15),

$$q^*(\omega) \;=\; \left(\frac{\beta(\omega)}{1+G_H(\omega)}\right) q^*(\omega) + \left(\frac{G_H(\omega)}{1+G_H(\omega)}\right) q_d(\omega). \tag{16}$$

or

$$q^*(\omega) = \left(\frac{G_H(\omega)}{1+G_H(\omega)-\beta(\omega)}\right) q_d(\omega) \tag{17}$$

resulting in the final output tracking error e^* from (7)

$$e^*(\omega) \;=\; q_d(\omega) - q^*(\omega) = \left(\frac{1-\beta(\omega)}{1-\beta(\omega)+G_H(\omega)}\right) q_d(\omega). \tag{18}$$

The human-response dynamics G_H impacts the output tracking error e as seen in (18). If the open-loop gain $|G_H(\omega)|$ is high, then good tracking can be achieved by the human alone. Therefore, the collaborative approach (where the robot does not know the desired output but aims to track some fraction of the achieved total human-robot output q_k) results in a small output tracking error e when the human tracking ability is good. Interestingly, even when the human tracking ability is not good, i.e., $|G_H(\omega)|$ is low, the final collaborative tracking can be good (i.e., a small output tracking error e can be achieved), if the assistance level is high, i.e., $\beta(\omega) \to 1$ since the numerator becomes close to zero in the right-hand-side of (18).

Remark 1 (Expert operator)**.** *When the desired output q_d is not known to the robot controller, the final tracking error e^* depends on the ability of the human to track the desired output, i.e., feedback response dynamics G_H of the human operator. Consequently, good final collaborative tracking (with different levels of assistance β is achieved when the human operator is an expert at tracking ($|G_H(\omega)|$ is high).*

Remark 2 (Novice operator)**.** *Even with a novice operator who does not have the expertise to achieve good tracking ($|G_H(\omega)|$ is low), and without direct access to the desired output q_d, good final tracking can be achieved with the proposed approach if the robotic assistance level is high, i.e., $\beta(\omega) \to 1$.*

Remark 3 (Frequency dependent assistance)**.** *Good tracking can often be achieved at low frequencies by human operators, but not at relatively high frequencies, which can be accommodated in the current formulation since it allows for a frequency-dependent, robotic assistance level $\beta(\omega)$.*

2.4 Iterative model-inversion algorithm

First, the ideal choice of the unknown model $\hat{G}_k^{-1}$ in (11) is investigated. Multiplying (11) throughout by the robot transfer function G_R, using (5) and suppressing frequency dependence on ω (for ease of notation), results in

$$q_{r,k+1} \;=\; q_{r,k} + \rho_k G_R \hat{G}_k^{-1} e_{r,k}. \tag{19}$$

Substituting for the output q_k from (15) into (10), the robot output $q_{r,k}$ at iteration step k can be expressed as

$$q_{r,k} \;=\; \left(\frac{\beta G_H}{1+G_H}\right) q_d - \left(\frac{1+G_H}{1+G_H-\beta}\right) e_{r,k}, \tag{20}$$

which can be used to substitute for $q_{r,k}$ and $q_{r,k+1}$ into (19), resulting in

$$\begin{aligned} e_{r,k+1} \;&=\; e_{r,k}\left[1-\hat{G}_k^{-1}\rho_k G_R\left(1-\frac{\beta}{1+G_H}\right)\right], \\ &=\; e_{r,k}\left[1-\hat{G}_k^{-1}\overline{G}\right], \end{aligned} \tag{21}$$

where, the transfer function, $\overline{G}$ is defined as,

$$\overline{G}(\omega) \;=\; \rho_k(\omega) G_R(\omega)\left(1-\frac{\beta}{1+G_H(\omega)}\right). \tag{22}$$

Ideally, the error $e_{r,k+1}$ at the next iteration step $k+1$ can be made zero, i.e., $e_{r,k+1} = 0$ in (21). However, this requires the selection of the model $\hat{G}_k$ in (21) as the transfer function $\overline{G}$, i.e.

$$\hat{G}_k(\omega) \;=\; \overline{G}(\omega). \tag{23}$$

However, no prior knowledge is available for this transfer function $\overline{G}$ (to use an a priori fixed model as in [23]), nor is input/output data available to compute $\overline{G}$ as in [25]. Therefore, the current article proposes the use of the error relation in (21) from the previous iteration step

$$e_{r,k} \;=\; e_{r,k-1}\left[1-\hat{G}_{k-1}^{-1}\overline{G}\right], \tag{24}$$

to estimate the transfer function $\overline{G}$ as

$$\overline{G}(\omega) \;=\; \hat{G}_{k-1}(\omega)\left(\frac{e_{r,k-1}(\omega)-e_{r,k}(\omega)}{e_{r,k-1}(\omega)}\right). \tag{25}$$

In practice, noise may be present in the measured output leading to noise n_k in the measured error $e_{r,k}$,

$$\tilde{e}_{r,k}(\omega) \;=\; e_{r,k}(\omega)+n_k(\omega). \tag{26}$$

Therefore, the model $\hat{G}_k$ in (23) is found using the model $\overline{G}$ computed using the measured errors, $\tilde{e}_{r,k}, \tilde{e}_{r,k-1}$, similar to (25), as

$$\hat{G}_k(\omega) \;=\; \hat{G}_{k-1}(\omega)\left(\frac{\tilde{e}_{r,k-1}(\omega)-\tilde{e}_{r,k}(\omega)}{\tilde{e}_{r,k-1}(\omega)}\right). \tag{27}$$

This estimate of the model $\hat{G}_k$ is used in the model-inversion based ILC algorithm in (11).

Remark 4 (Update dependence on models)**.** *While the estimated model $\hat{G}_k$ update in (27) is based on the error term $e_{r,k-1}$ and $e_{r,k}$, it aims to match $\overline{G}$ as in (23). Note that $\overline{G}$ depends on both the human-response dynamics G_H and the robot dynamics G_R as seen in (22).*

2.5 Convergence of proposed ILC

The model update in (27) leads to convergence of the collaborative ILC in (11) under the following assumption that the noise term n_k in (26) is bounded from above for each frequency ω and for each iteration step k as,

$$|n_k(\omega)| \;<\; N(\omega) \;<\; \infty. \tag{28}$$

Lemma (Convergence)**.** *The collaborative update law in (11) with the model update in (27) results in convergence of the robot tracking error $e_{r,k}$ (without measurement noise) at frequency ω, i.e.,*

$$\lim_{k\to\infty}|e_{r,k}(\omega)| \;=\; 0, \tag{29}$$

if for some iteration step k^* the following conditions are met, the measured error is sufficiently larger than the noise, i.e.,

$$|\tilde{e}_{r,k^*-1}(\omega)| \;>\; \alpha_1 N(\omega), \tag{30}$$

the difference in the measured error is sufficiently larger than the noise, i.e.,

$$|\tilde{e}_{r,k^*}(\omega)-\tilde{e}_{r,k^*-1}(\omega)| \;>\; \alpha_2 N(\omega), \tag{31}$$

where α_1, α_2 are positive scalars that satisfy,

$$\frac{2\alpha_1+\alpha_2}{\alpha_2(\alpha_1-1)} \;<\; 1, \tag{32}$$

and the model update is given by

$$\hat{G}_k^{-1}(\omega) \;=\; \begin{cases} \hat{G}_{k-1}^{-1}(\omega)\left(\frac{\tilde{e}_{r,k-1}(\omega)}{\tilde{e}_{r,k-1}(\omega)-\tilde{e}_{r,k}(\omega)}\right), \\ \quad \textit{when (30) and (31) are satisfied,} \\ \\ \hat{G}_{k-1}^{-1}(\omega), \textit{otherwise.} \end{cases} \tag{33}$$

Proof. The error-dynamics (without measurement noise) in (21), results in a contraction leading to the convergence condition in (29) if the error gain term, $\gamma_k = 1-\hat{G}_k^{-1}\overline{G}$, in (21) has magnitude less than one, i.e., there exists some iteration step k^* beyond which, i.e., for $k \geq k^*$,

$$|\gamma_k(\omega)| \;=\; \left|1-\hat{G}_k^{-1}(\omega)\overline{G}(\omega)\right| \;<\; 1. \tag{34}$$

Using (25) and (27) in the LHS of the inequality in (34), and suppressing the dependence on ω for ease in notation,

$$\begin{aligned} |\gamma_k| &= \left|1-\left(\tfrac{\tilde{e}_{r,k-1}}{\tilde{e}_{r,k-1}-\tilde{e}_{r,k}}\right)\left(\tfrac{e_{r,k-1}-e_{r,k}}{e_{r,k-1}}\right)\right| \\ &\quad \text{and using (29),} \\ &= \left|1-\left(\tfrac{e_{r,k-1}+n_{k-1}}{e_{r,k-1}}\right)\left(\tfrac{\tilde{e}_{r,k-1}-\tilde{e}_{r,k}+n_k-n_{k-1}}{\tilde{e}_{r,k-1}-\tilde{e}_{r,k}}\right)\right| \\ &= \left|\tfrac{n_{k-1}}{e_{r,k-1}}+\tfrac{n_k-n_{k-1}}{\tilde{e}_{r,k-1}-\tilde{e}_{r,k}}+\left(\tfrac{n_{k-1}}{e_{r,k-1}}\right)\left(\tfrac{n_k-n_{k-1}}{\tilde{e}_{r,k-1}-\tilde{e}_{r,k}}\right)\right| \\ &\quad \text{and applying the triangle inequality,} \\ &\leq \left|\tfrac{n_{k-1}}{e_{r,k-1}}\right|+\left|\tfrac{n_k-n_{k-1}}{\tilde{e}_{r,k-1}-\tilde{e}_{r,k}}\right|+\left|\left(\tfrac{n_{k-1}}{e_{r,k-1}}\right)\left(\tfrac{n_k-n_{k-1}}{\tilde{e}_{r,k-1}-\tilde{e}_{r,k}}\right)\right| \end{aligned} \tag{35}$$

Each term in the last expression can be bounded using the measurement noise bound in (28) as,

$$\left|\frac{n_{k-1}}{e_{r,k-1}}\right| \;=\; \left|\frac{n_{k-1}}{\tilde{e}_{r,k-1}-n_{k-1}}\right| \;\leq\; \frac{N}{(\alpha_1-1)N} = \frac{1}{\alpha_1-1}, \tag{36}$$

and,

$$\left|\frac{n_k-n_{k-1}}{\tilde{e}_{r,k-1}-\tilde{e}_{r,k}}\right| \;\leq\; \frac{2N}{\alpha_2 N} \;=\; \frac{2}{\alpha_2}. \tag{37}$$

Then, applying the bounds in (36) and (37) to the expression in (35) results in

$$\begin{aligned} |\gamma_k(\omega)| \;&\leq\; \frac{1}{\alpha_1-1}+\frac{2}{\alpha_2}+\frac{2}{\alpha_2(\alpha_1-1)} \\ &=\; \frac{2\alpha_1+\alpha_2}{\alpha_2(\alpha_1-1)} \;<\; 1, \end{aligned} \tag{38}$$

which is the required condition in (32) of the Lemma. This completes the proof. □

Remark 5. *The required condition in* (32) *is satisfied if the constants* α_1, α_2 *are chosen large enough, e.g.,*

$$\alpha_1 > 2, \quad \alpha_2 \; > \frac{2\alpha_1}{\alpha_1 - 2} \tag{39}$$

where the second inequality follows from (32) *if* $\alpha_1 > 2$ *and* $\alpha_2 > 0$ *since* $2\alpha_1 + \alpha_2 < \alpha_2\alpha_1 - \alpha_2$.

3 System description and implementation

The experiment was selected to represent applications where the robot and the human operator are collaboratively completing a task, e.g., (i) in applications such as rehabilitation where the robot might be programmed by kinesthetic teaching with the human operator physically (collaboratively) moving the endpoint of the robot, and (ii) in manufacturing operations where the operator might be moving a tool with assistance from the robot. The goal is to allow novice operators, who might be good at the task but not good at operating the robot, can work with the robot. Towards this, the human-in-the-loop response dynamics is modeled and corrected iteratively as illustrated with the experimental system described below.

3.1 Overview of experimental setup

A MICO2 robot arm, manufactured by Kinova Robotics was used in the human-robot experiments, where the objective was to follow a specified output trajectory, known to the human operator but not known to the robot. Specifying the desired output $\mathbf{q}_d$ to the human operator allowed the evaluation of the convergence with the proposed iterative approach and quantification of the error in output tracking. The experimental setup is shown in Fig. 2.

A flexible structure consisting of a coil spring was attached rigidly to the end-effector of the MICO2 robot arm. A laser attached to the end of the flexible structure projected a red dot on to the screen — the position of the red dot on the screen was the output vector,

$$\mathbf{q} = [q_x, q_y]^T. \tag{40}$$

In what follows, vectors will be denoted in bold; non-bold symbols, such as q, will denote scalars in either the x or y axes, with the direction specifically clarified with a subscript, e.g., q_x, when necessary. An LCD projector displayed the desired trajectory $\mathbf{q}_d$, represented by the center of a green circle. A camera was used to measure the real-time positions of the laser dot $\mathbf{q}$ and the green circle $\mathbf{q}_d$ at a frame rate of 30Hz and the signals were filtered with a zero-phase second-order Butterworth filter with cutoff frequency of 5 Hz [34]. Image processing tools used, e.g., to find the center of the green circle for the desired output $\mathbf{q}_d$, were from the OpenCV library [35]. The objective was to track the center of the green circle $\mathbf{q}_d$, whose trajectory was specified but not known a priori to the human operator (and was not available to the robot controller), using the red laser dot $\mathbf{q}$.

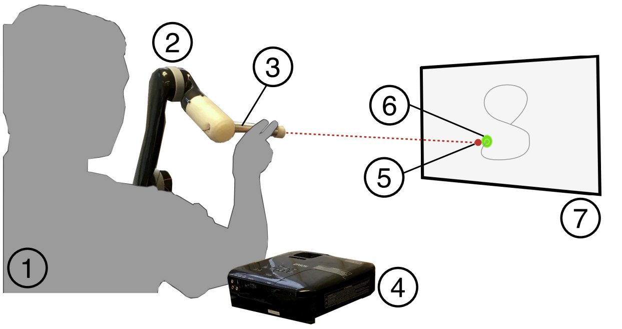

Fig. 2 Experimental setup for the human-robot output-tracking experiment: (1) human operator; (2) Kinova Mico 4-dof robot arm; (3) flexible end-effector with laser pointer embedded in the tip; (4) LCD projector used to display the desired output $\mathbf{q_d}$ (green circle) (6) and Logitech C920 web-camera (positioned adjacent to the projector (not shown)) used to sense the output $\mathbf{q}$ (laser) (5) and the desired output $\mathbf{q_d}$ (green circle) (6); and (7) screen where projected image is displayed.

3.2 Flexible human-robot interface

The proposed collaborative human-robot control scheme was applied through a flexible-spring system shown in Fig. 3. The flexibility of the spring human-robot-interface allowed both the human and robot to simultaneously control the total output, $\mathbf{q}$, namely the position of the red dot on the screen This facilitated demonstration of the task (such as tracking the desired output) by the human operator even when the robot was operating independently, e.g., during the initial trial $k = 0$ when the robot had no information about the desired output.

The robot end-effector was attached to one end of the flexible structure, i.e., end (A) in Fig. 3, while the human operator deflected the structure relative to end (A) by applying a displacement, δ_h at the other end, i.e., end (B) in Fig. 3. Thus, relative to the end (A), the flexible structure can be considered as a cantiliver beam with a concentrated load, P acting at end (B) due to the human operator. Then, the displacement of end (B) relative to end (A), i.e., δ_h can be related to the slope, θ as, (using the Timoshenko beam theory [36])

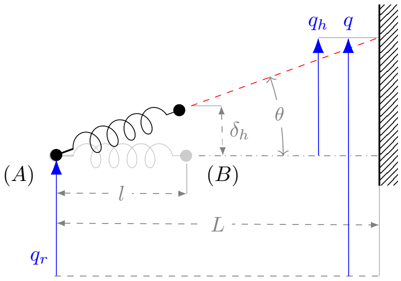

Fig. 3 Collaborative control of the flexible structure. (a) Robot end-effector controlled position, $q_r = (x_r, y_r)$ of end (A), (b) Human operator's deflection with respect to end (A), $\delta_h = (\delta_{x,h}, \delta_{y,h})$, (c) Total position, $q = q_r + q_h$, where $q_h = f(\delta_h)$, as in (42).

$\theta \approx \frac{3\delta_h}{2l}$, where, l is the length of the flexible structure, and $\delta_h = (\delta_{h,x}, \delta_{h,y})$ are deflections of end (B) along each coordinate axis (x, y). The position q of the red dot on the screen can then be obtained (in either the x or y axes) as,

$$q = q_r + q_h, \quad \text{where } q_h = f(\delta_h). \tag{41}$$

When the distance L to the screen is large compared to the length l of the flexible structure, i.e., $L \gg l$, the relationship between the deflection, δ_h and the screen position, q_h can be simplified to,

$$q_h = f(\delta_h) \approx L \tan\theta = L \tan\left(\frac{3\delta_h}{2l}\right) \approx L\left(\frac{3\delta_h}{2l}\right), \tag{42}$$

where the angle θ is as illustrated in Fig. 3.

3.3 Screen To Robot Transformation

The screen where the position of the robot is measured was not fully parallel to the x, y axes of the robot end effector motion. Hence a transformation was used to map the observed positions on the screen $\mathbf{q}^S$ to the robot x, y coordinates $\mathbf{q}^R$:

$$\mathbf{q}^R = \mathbf{A}\mathbf{q}^S + \mathbf{b} \tag{43}$$

where the elements of the 2×2 matrix $\mathbf{A}$ and 2×1 vector $\mathbf{b}$ were found using a least squares fit with experimentally measured $\mathbf{q^R}$ and $\mathbf{q^S}$. Note that this map is invertible, i.e., $\mathbf{A}$ is invertible. In the following, most data are presented in the robot coordinate frame. Therefore, the superscript R in (43) for the robot frame is not added for notational ease. However, the superscript S for the screen frame is explicitly denoted.

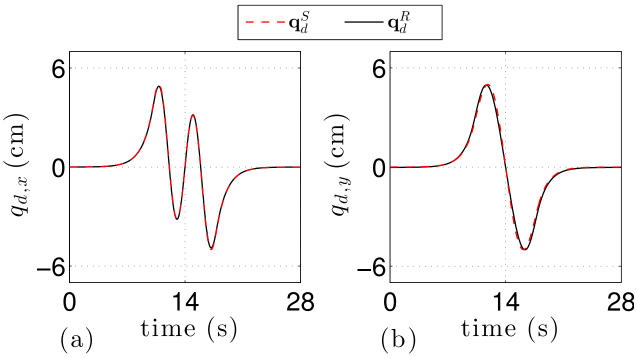

Fig. 4 Desired output $\mathbf{q}_d = [q_{d,x}, q_{d,y}]^T$ in Section 3.5 in robot frame $\mathbf{q}_d^R = [q_{d,x}^R, q_{d,y}^R]^T$ and screen frame. $\mathbf{q}_d^S = [q_{d,x}^S, q_{d,y}^S]^T$.

3.4 Robot position control

The velocity of the robot arm's end-effector can be directly controlled in Cartesian space with an Application Programming Interface (API), written in C++. In this mode of operation, the linear speed of the robot was limited to 0.2 (m/s), which aids a human operator to safely interact with the robot. An outer proportional feedback loop was designed around the inner Kinova velocity controller. Thus, the controller was designed to track the robot command $\mathbf{u} = [u_x, u_y]^T$, which resulted in the robot position $\mathbf{q}_r = [q_{r,x}, q_{r,y}]^T$.

3.5 The desired output trajectory

The desired output trajectory $\mathbf{q}_d$ is shown in Fig. 4. To design the desired output, first, unfiltered nominal output components x_n, y_n were specified as

$$x_n(t) = \begin{cases} \sin[2\omega_0(t - t_1)] & \text{if} \quad t \in [t_1, t_2] \\ 0 & \text{if} \quad t \in [0, t_1] \cup [t_2, T_p] \end{cases} \tag{44}$$

$$y_n(t) = \begin{cases} \sin(\omega_0(t - t_1)) & \text{if} \quad t \in [t_1, t_2] \\ 0 & \text{if} \quad t \in [0, t_1] \cup [t_2, T_p] \end{cases} \tag{45}$$

where the frequency $f_0 = \omega_0/2\pi$ was selected as $f_0 = 0.125$ (Hz), the movement occurred over one time period of the slower sinusoid, i.e., $t_2 - t_1 = T_0 = 1/f_0 = 8$ (s), and zero-displacement padding of 10 (s) was added at the beginning and end, i.e., $t_1 = 10$ (s) and the total time was $T_p = 28$ (s). To ensure that the desired output $\mathbf{q}_d^S = [q_{d,x}^S, q_{d,y}^S]^T$ (in the screen frame) is sufficiently smooth, the nominal output components x_n, y_n were filtered and scaled as,

$$\begin{aligned} q_{d,x}^S(\omega) &= s_1 G_f(\omega) x_d(\omega) \\ q_{d,y}^S(\omega) &= s_2 G_f(\omega) y_d(\omega), \end{aligned} \tag{46}$$

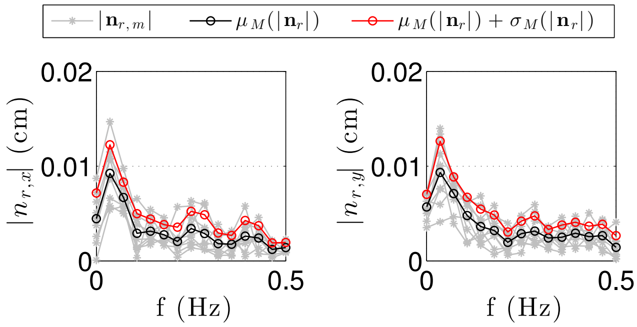

Fig. 5 Estimated frequency-dependent upper bounds $N_x(\omega)$ and $N_y(\omega)$ on noise defined in (50).

with a zero phase filter G_f,

$$G_f(\omega) = \left(\frac{\omega_c}{\omega_c - j\omega}\right)\left(\frac{\omega_c}{\omega_c + j\omega}\right) \tag{47}$$

where $j = \sqrt{-1}$, the cutoff frequency $f_c = \omega_c/2\pi = 0.15$ (Hz), and $s_1 > 0, s_2 > 0$ in (46) were real-valued scaling factors used to obtain a peak-to-peak amplitude of 10 (cm) in each component of the desired output trajectory $\mathbf{q}_d^S$ as in Fig. 4. A measure of the size of the desired output $\mathbf{q}_d$ is

$$q_{d,max} = \max_{t\in[0,T_p]} \sqrt{q_{d,x}^S(t)^2 + q_{d,y}^S(t)^2} = 6.5 \text{ (cm)}, \tag{48}$$

which is used in the following to normalize the measured tracking error.

3.6 Algorithm implementation

Algorithm 1 Data-driven ILC

```
Require: $\beta, \alpha_1, \alpha_2, N(\omega), \rho(\omega), \hat{G}_0^{-1}(\omega), u_0(\omega)$
Initialize:
Apply control $u_0(\omega)$, measure outputs $q_0(\omega)$, $q_{r,0}(\omega)$.
Compute: $\tilde{e}_{r,0} = \beta q_0(\omega) - q_{r,0}(\omega)$ (10).
Update: $u_1(\omega) = u_0(\omega) + \rho(\omega)\hat{G}_0^{-1}(\omega)\tilde{e}_{r,0}(\omega)$ (11).
$k = 1$.
while iteration step $k < k^*$ with $k^* = 20$ in current experiments do
    Apply control $u_k(\omega)$, measure outputs $q_k(\omega)$, $q_{r,k}(\omega)$.
    Compute: $\tilde{e}_{r,k} = \beta q_k(\omega) - q_{r,k}(\omega)$ (10).
    for each frequency $\omega$ do
        Update model:
        if $|\tilde{e}_{r,k}(\omega)| > \alpha_1 N(\omega)$ (30)
         and $|\tilde{e}_{r,k}(\omega) - \tilde{e}_{r,k-1}(\omega)| > \alpha_2 N(\omega)$ (31) then
            $\hat{G}_k^{-1}(\omega) = \hat{G}_{k-1}^{-1}(\omega)\left(\frac{\tilde{e}_{r,k-1}(\omega)}{\tilde{e}_{r,k-1}(\omega) - \tilde{e}_{r,k}(\omega)}\right)$ (33),
        else
            $\hat{G}_k^{-1}(\omega) = \hat{G}_{k-1}^{-1}(\omega)$ (33).
        end if
    end for
    Update:
    $u_{k+1}(\omega) = u_k(\omega) + \rho(\omega)\hat{G}_k^{-1}(\omega)\tilde{e}_{r,k}(\omega)$ (11).
    $k = k + 1$.
end while
```

To implement the proposed algorithm, described in one dimension in Algorithm 1, at each iteration step k, input $\mathbf{u}_k$ was applied to the robot, and the resulting output $\mathbf{q}_k^S$ was measured and mapped into the robot's coordinate frame using (43). Also, the inverse model $\hat{G}_k^{-1}(\omega)$ was computed (in both (x,y) coordinates) using (33), and the robot command input $\mathbf{u}_{k+1}$ was computed using (11) and applied to the robot on the following iteration step $k+1$. The model update was performed when the estimated error was sufficiently large compared to the measurement noise $\mathbf{n}_k$. Towards this, the constants α_1, α_2 in (32) were selected to be $\alpha_1 = 15, \alpha_2 = \alpha_1/5$ and an estimate on the upper bound $N(\omega)$ of the noise in (30) was experimentally selected, based on the robot noise, as follows. The robot input $\mathbf{u}$ was repeatedly commanded (M times) to track a fixed trajectory and the deviations in the achieved trajectory was used to estimate the controller-related system noise. Although the robot controller does not have access to the desired output during human-robot collaboration experiments, the desired output $[q_{d,x}, q_{d,y}]^T$ (already described earlier) was selected as the nominal trajectory for the noise estimation. This allows the comparative evaluation of the results of the experiments in terms of noise at the desired trajectory. In practice, when the desired trajectory is unknown, other trajectories could be used to estimate the noise. The noise at the m^{th} trial was characterized as the deviation from the mean value of the data, i.e.,

$$\mathbf{n}_{r,m}(t) = \mathbf{q}_{r,m}(t) - \bar{\mathbf{q}}_r(t) \tag{49}$$

where $\bar{\mathbf{q}}_r(t)$ denotes the mean value at time t over the M trials. Then the upper bound on the noise, shown in Fig. 5, in the x and y axes were estimated as

$$\begin{aligned} N_x(\omega) &= \mu_M(|n_{r,x}(\omega)|) + \sigma_M(|n_{r,x}(\omega)|) \\ N_y(\omega) &= \mu_M(|n_{r,y}(\omega)|) + \sigma_M(|n_{r,y}(\omega)|) \end{aligned} \tag{50}$$

with the pointwise (in frequency ω) mean μ_M and standard deviation σ_M defined as

$$\mu_M(f) = \frac{1}{M}\sum_{m=1}^{M} f_m(\cdot), \tag{51}$$

$$\sigma_M(f) = \frac{1}{M}\sum_{m=1}^{M} \left(f_m(\cdot) - \mu_M(f)\right)^2, \tag{52}$$

where f is a dummy variable. The learning gain $\rho_k(\omega)$ in (11) (for both axes) was limited to the learning bandwidth, i.e.,

$$\rho_k(\omega) = \begin{cases} \rho & \text{if } \omega \le \omega_{max} \\ 0 & \text{otherwise,} \end{cases} \tag{53}$$

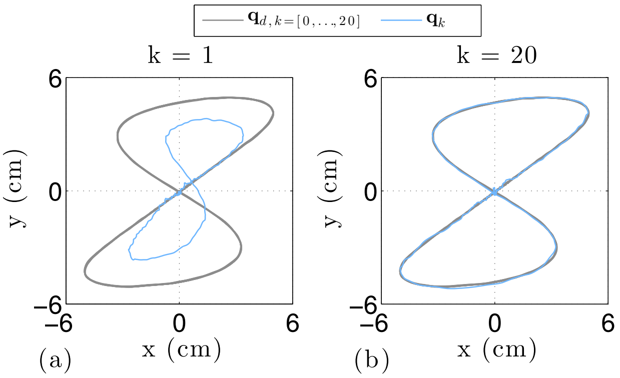

Fig. 6 (Robot-only case) Tracking results in (x, y) coordinates for iteration steps: (a) $k = 1$, and (b) $k = 20$ during the verification experiment with $\rho = 1$. Desired output $\mathbf{q}_{d,k}$, defined in (46), for all iteration steps k are plotted, and the thickness of the plot represents the camera noise.

where ρ is a non-negative scalar. The nominal value of the iteration gain was selected as $\rho = 1$ and the maximum learning frequency was selected as $f_{max} = \omega_{max}/2\pi = 0.5$ (Hz), which is near the bandwidth limit of human smooth visual tracking [16]. The inverse model in (33) was initialized to unity for each frequency, e.g., $\hat{G}_0^{-1}(\omega) = 1$. Note that, when available, nominal models averaged over different human operators, or models acquired from previous trials for the individual human operator could be used as the initial inverse model $\hat{G}_0^{-1}$.

4 Results and discussion

Experimental results are presented to evaluate the tracking performance of the proposed ILC algorithm. Results are discussed for two cases: (i) skill transfer case when the robot is fully tracking the desired output ($\beta = 1$) and (ii) co-tracking case when the human and the robot are collaborating ($\beta = 0.5$) on the output tracking task. The section begins by presenting results for verification of the proposed approach with only the robot, without the human.

4.1 Verification with robot-only case

The proposed algorithm was implemented to track the desired output $\mathbf{q}_d$ (see Fig. 4) *without* the human operator to verify if precision output tracking can be achieved, and whether the model estimated using the proposed error-based algorithm matched the expected robot dynamics. Note that without the human in the verification trials, the total output $\mathbf{q}_k$ at iteration step k was just the robot output, i.e., $\mathbf{q}_k = \mathbf{q}_{r,k}$ (see (4)), and the desired output $\mathbf{q}_{d,k}$, at iteration step k, was directly supplied to the ILC algorithm, as in (8) and (9).

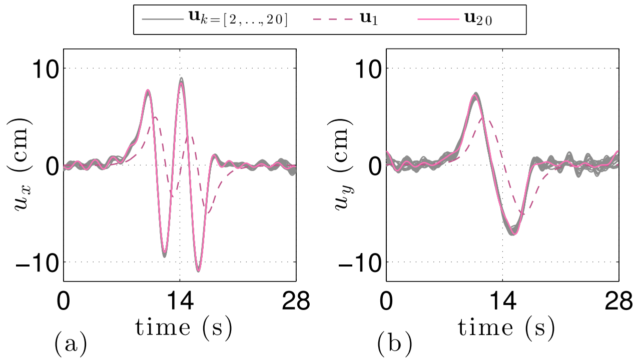

Fig. 7 (Robot-only case) Learned control input during verification experiment with learning gain $\rho = 1$: (a) $u_{x,k}$ and (b) $u_{y,k}$.

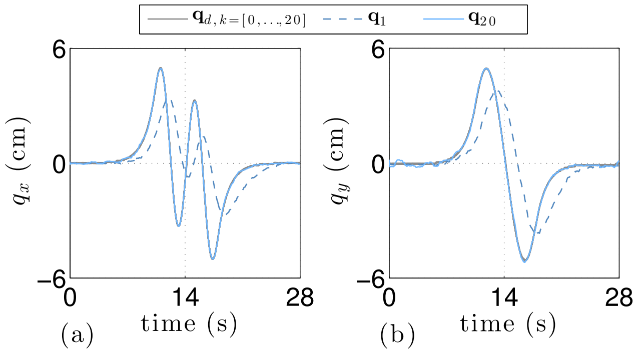

Fig. 8 (Robot-only case) Tracking results in time during the verification experiment with learning gain $\rho = 1$: (a) $q_{x,k}$, and (b) $q_{y,k}$.

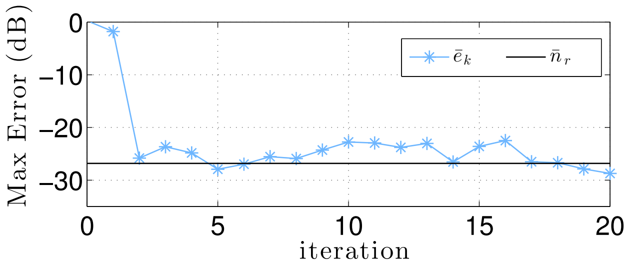

Fig. 9 (Robot-only case) Tracking error $\bar{e}_k$, defined in (54), for each iteration step during the verification experiment. Robot noise $\bar{n}_r$, defined in (55), denoted as the solid horizontal line.

4.1.1 Achieved output tracking

The proposed algorithm *without* the human operator achieved precision output tracking, reducing the tracking error close to the robot noise level. The tracking

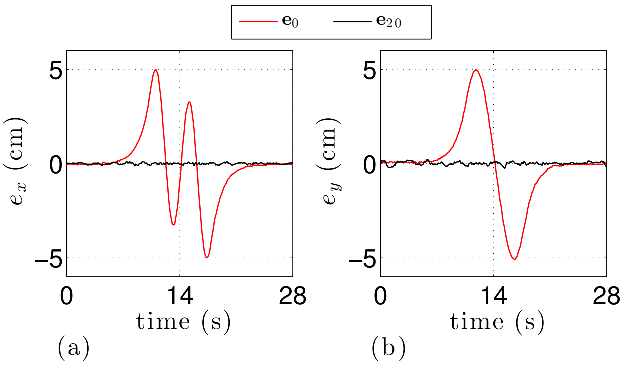

Fig. 10 (Robot-only case) Tracking error $\mathbf{e}_k$, defined in (7), in time during verification experiment for iteration steps $k = 0$ and $k = 20$: (a) $e_{x,k}$ and (b) $e_{y,k}$.

error at each iteration step k was characterized as,

$$\bar{e}_k(\%) = \max_{t\in[0,T_p]} \frac{\sqrt{e_{x,k}(t)^2 + e_{y,k}(t)^2}}{q_{d,max}} \times 100 \tag{54}$$

where $e_{x,k}$ and $e_{y,k}$ were the components of the error defined in (7). The control input $\mathbf{u}_1$ at iteration step $k = 1$, seen in Fig. 7, led to a large phase error in the output $\mathbf{q}_1$ (Fig. 8). However, by the second $k = 2$ iteration step, this phase error had been corrected substantially, and by the last iteration step, shown in the (x, y)-space in Fig. 6(b) and as function of time in Fig. 8, the robot tracking error reduced to $\bar{e}_{20} = 3.7\%$. The tracking error quantified as $\bar{e}_k$ and shown in Fig. 9 decreased with iteration step k and eventually converged close to the robot noise $\bar{n}_r$, characterized with M repeated commands as in Section 3.6, and computed as

$$\bar{n}_r(\%) = \max_{\substack{m\in M\\ t\in[0,T_p]}} \frac{\sqrt{n_{r,x,m}(t)^2 + n_{r,y,m}(t)^2}}{q_{d,max}} \times 100 \tag{55}$$

where $\mathbf{n}_{r,m}$ defined in (49), and the maximum is taken over both time and M trials. The robot noise $\bar{n}_r$ provided an estimate of the lower bound on robot performance and found to be 4.6% of the desired output. Fig. 10 shows each component of the error $\mathbf{e}_k$, defined in (7), in the time domain for iteration steps $k = 0$ and $k = 20$. The results of the verification experiment are recorded in row 1, and the noise value in row 4 of Table 1.

The proposed model update law in (33) captured the robot dynamics during the verification (robot-only) experiment. Figure 11 shows the last iteratively learned model $\hat{G}_{x,20}(\omega)$ (e.g., the inverse of (33)) and compares it with the robot dynamics $G_{R,x}(\omega)$ computed as

$$G_{R,x}(\omega) = \frac{q_{x,20}(\omega)}{u_{x,20}(\omega)} \tag{56}$$

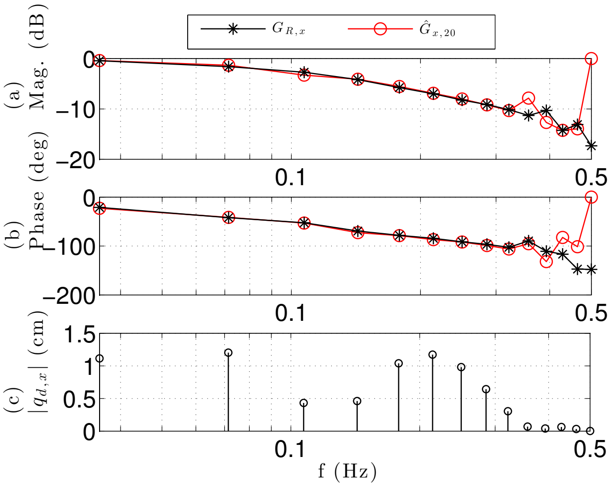

Fig. 11 Iteratively learned model $\hat{G}_{x,20}$, defined in (33), compared with calculated robot dynamics $G_{R,x}$, defined in (56): (a) magnitude reported in decibels, (b) phase in degrees, and (c) spectrum of the desired output $q_{d,x}$, defined in (46), for all k iteration steps.

at the $k = 20$ iteration step. From the spectrum of the desired output shown in Fig. 11(c), it can been seen that the learned model $\hat{G}_{x,20}(\omega)$ matched the robot dynamics $G_{R,x}(\omega)$ at frequencies where the desired output spectrum had significant content, up to ∼ 0.35 Hz.

4.2 Skill transfer case, $\beta = 1$

The proposed method, applied to the human-robot collaboration experiment with level of assistance $\beta = 1$, where the robot aims to track the output, resulted in tracking well below the human noise $\bar{n}_h$, and tracking error $\bar{e}_{20} = 10.0\%$. The human noise $\bar{n}_h$, used as an estimate of expected human performance without the robot collaborator, was quantified as the maximum error during the initial human demonstration, e.g., at iteration step $k = 0$:

$$\bar{n}_h(\%) = \max_{t\in[0,T_p]} \frac{\sqrt{e_{x,0}(t)^2 + e_{y,0}(t)^2}}{q_{d,max}} \times 100. \tag{57}$$

The achieved output tracking is shown in Fig. 12 for a few iteration steps. During the initial $k = 0$ iteration step (Fig. 12(a)), the initial human demonstration can be seen as the output $\mathbf{q}_k(t)$ (e.g., $\mathbf{q}_r(t) = 0$, see (4)). By iteration step $k = 3$, shown in Fig. 12(c), the robot had learned most of the desired trajectory, evidenced by the large reduction in human output $q_h(t)$. The robot tracking error was quantified as

$$\bar{e}_{d,k}(\%) = \max_{t\in[0,T_p]} \frac{\sqrt{e_{d,x,k}(t)^2 + e_{d,y,k}(t)^2}}{q_{d,max}} \times 100 \tag{58}$$

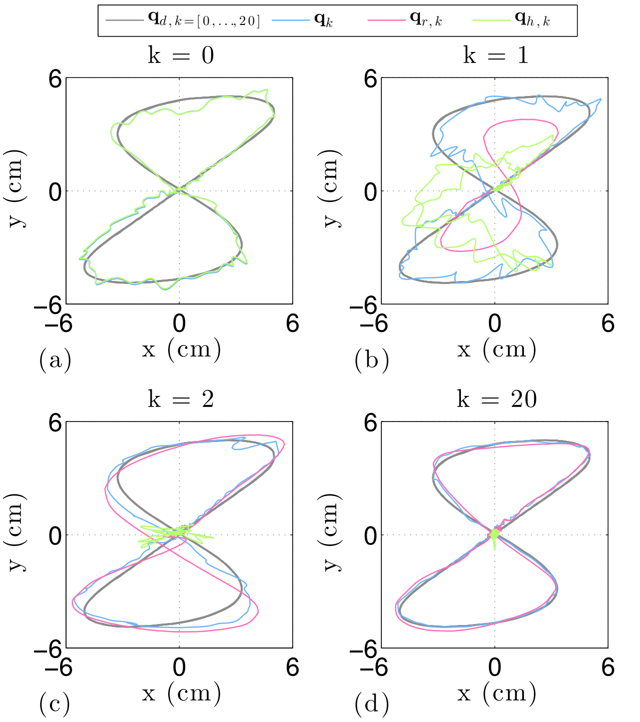

Fig. 12 Tracking results in (x, y) coordinates during human-robot collaboration experiment with level of assistance $\beta = 1$ and learning gain $\rho = 1$, for iteration steps: (a) $k = 0$, (b) $k = 1$, (c) $k = 3$, and (d) $k = 20$. Desired output $\mathbf{q}_{d,k}(t)$, defined in (46), for all iteration steps k are plotted, and the thickness of the plot represents the camera noise.

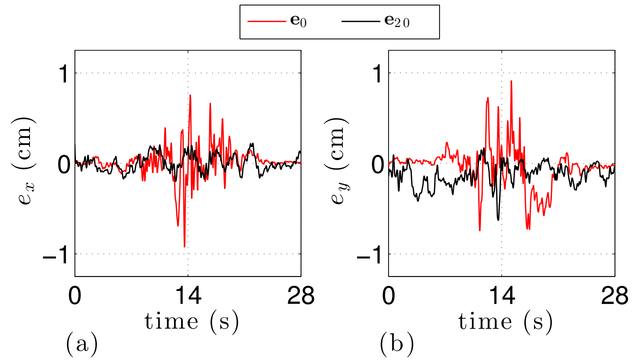

Fig. 13 Tracking error $\mathbf{e}_k$, defined in (7), in time during the human-robot collaboration experiment with level of assistance $\beta = 1$ and learning gain $\rho = 1$ for iteration steps $k = 0$ and $k = 20$: (a) $e_{x,k}$ and (b) $e_{y,k}$.

with $e_{d,x,k}$ and $e_{d,y,k}$ were the components of the error defined in (9). At the $k = 20$ iteration step (Fig. 12(d)), the robot tracked the desired output $\mathbf{q}_d$ with a robot tracking error $\bar{e}_{d,20} = 12.7\%$, and the human output $\mathbf{q}_h$ nearly vanished. Reduction in tracking error is observed in Figure 13, which shows each component of the error $\mathbf{e}_k$, defined in (7), and in Fig. 14, which shows each component of the robot tracking error $\mathbf{e}_{d,k}$, defined in (9), in the time domain, for iteration steps $k = 0$ and $k = 20$.

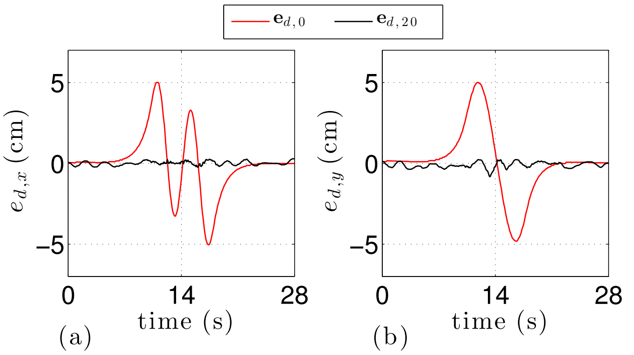

Fig. 14 Robot tracking error $\mathbf{e}_{d,k}$, defined in (9), in time during the human-robot collaboration experiment with level of assistance $\beta = 1$ and learning gain $\rho = 1$ for iteration steps $k = 0$ and $k = 20$: (a) $e_{d,x,k}$ and (b) $e_{d,y,k}$.

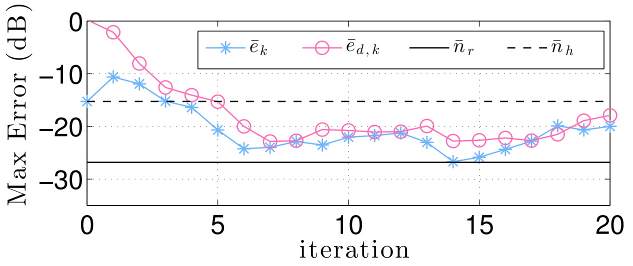

Fig. 15 Tracking error $\bar{e}_k$, defined in (54), and robot tracking error $\bar{e}_{d,k}$, defined in (58), for each iteration step during the human-robot collaboration experiment with level of assistance $\beta = 1$ and learning gain $\rho = 1$. The robot noise $\bar{n}_r$, defined in (55), and human noise $\bar{n}_h$, defined in (57), can be seen as the solid and dashed horizontal lines, respectively.

The tracking error $\bar{e}_k$, defined in (54), reduced to just greater than twice the size of the robot noise during the human-robot collaboration as seen in Fig. 15.

4.3 Co-tracking case, $\beta = 0.5$

Choosing the level of assistance $\beta = 0.5$ resulted in the robot providing approximately half of the desired output, and the human provided the remaining half. Figure 16 shows tracking results for sample iteration steps. The initial human demonstration is shown in Fig. 16(a); note that the scaled version of the desired output $\beta\mathbf{q}_{d,k}$ can be seen as the solid black traces. During the $k = 1$ iteration step (Fig. 16(b)), the robot provided some of the scaled desired output. By the final iteration step ($k = 20$), shown in Fig. 16(d), the robot output $\mathbf{q}_r$ tracked about half of the desired output and the human output $\mathbf{q}_h$ was reduced to half. Figure 17 shows each component of the error $\mathbf{e}_k$, defined in (7), and Fig. 18 shows each component of the robot tracking error $\mathbf{e}_{d,k}$, defined in (9), in the time domain for iteration

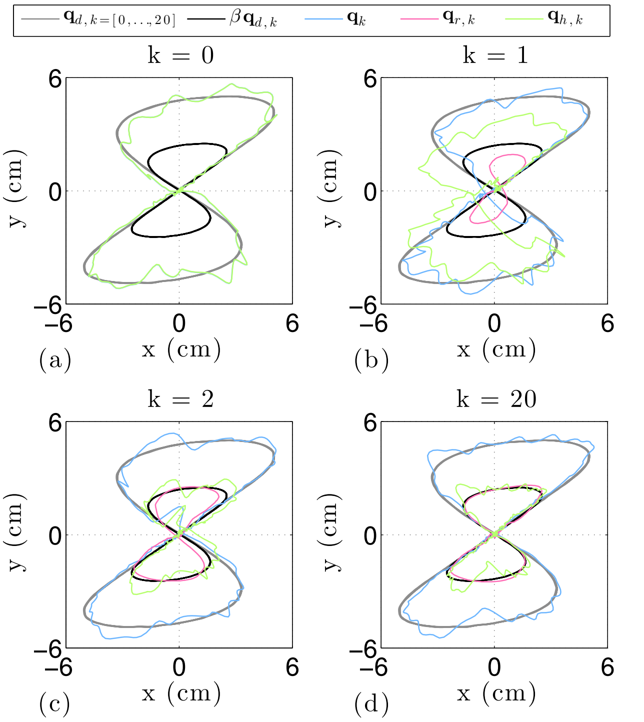

Fig. 16 Tracking results in (x, y) coordinates during human-robot collaboration experiment with level of assistance $\beta = 0.5$ and learning gain $\rho = 1$, for iteration steps: (a) $k = 0$, (b) $k = 1$, (c) $k = 3$, and (d) $k = 20$. Desired output $\mathbf{q}_{d,k}(t)$, defined in (46), for all iteration steps k are plotted, and the thickness of the plot represents the camera noise.

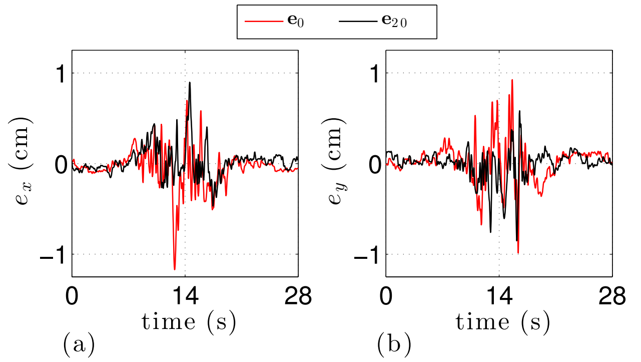

Fig. 17 Tracking error $\mathbf{e}_k$, defined in (7), in time, during the human-robot collaboration experiment with level of assistance $\beta = 0.5$ and learning gain $\rho = 1$ for iteration steps $k = 0$ and $k = 20$: (a) $e_{x,k}$ and (b) $e_{y,k}$.

steps $k = 0$ and $k = 20$. The reduction of the error ($\bar{e}_k$, defined in (54), and robot tracking error $\bar{e}_{d,k}$, defined in (58)) with iteration step k are shown in Fig. 19. During the final $k = 20$ iteration step, the tracking error $\bar{e}_{20}$ and the robot tracking error $\bar{e}_{d,20}$ were 16.6% and 4.6%, respectively, as summarized in Table 1.

The final output tracking error with the collaborative tracking $\beta = 0.5$ was larger than the error for the skill-transfer case with a larger level of assistance $\beta = 1$, see Fig. 12 for comparison. This is expected since according to (18), the final tracking error depends on the human performance when the level of assistance β is less than one, i.e., $\beta \neq 1$ since the human must generate some of the desired output $\mathbf{q}_d$. On the other hand, with the level of assistance set at $\beta = 1$, the human's

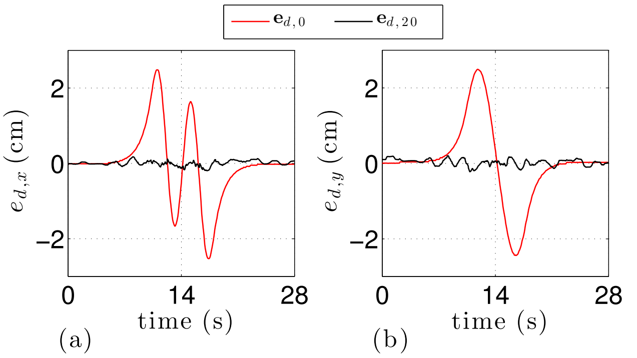

Fig. 18 Robot tracking error $\mathbf{e}_{d,k}$, defined in (9), in time during the human-robot collaboration experiment with level of assistance $\beta = 0.5$ and learning gain $\rho = 1$ for iteration steps $k = 0$ and $k = 20$: (a) $e_{d,x,k}$ and (b) $e_{d,y,k}$.

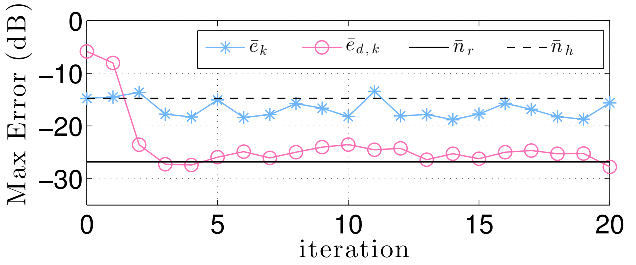

Fig. 19 Tracking error $\bar{e}_k$, defined in (54), and robot tracking error $\bar{e}_{d,k}$, defined in (58), for each iteration step during the human-robot collaboration experiment with level of assistance $\beta = 0.5$ and learning gain $\rho = 1$. The robot noise $\bar{n}_r$, defined in (55), and human noise $\bar{n}_h$, defined in (57), can be seen as the solid and dashed horizontal lines, respectively.

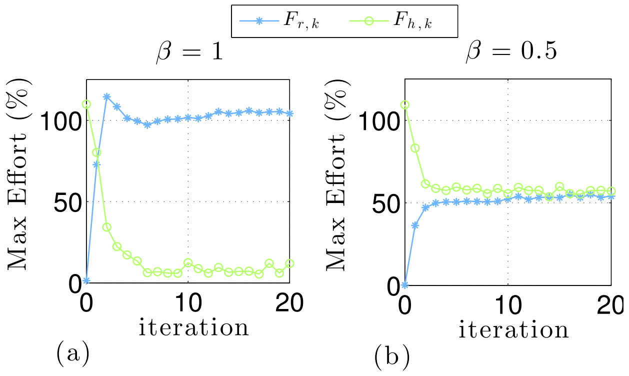

Fig. 20 Robot effort $F_{r,k}$ and human effort $F_{h,k}$, defined in (59), for each iteration step k, during: (a) skill-transfer case, and (b) co-tracking case.

Table 1 Summary of experimental results. Tracking error $\bar{e}_k$ is defined in (54), robot tracking error $\bar{e}_{d,k}$ is defined in (58), robot noise $\bar{n}_r$, reported in row 4, is defined in (55), and human noise $\bar{n}_h$, in row 5, is defined in (57).

		$\bar{e}_{20}(\%)$	$\bar{e}_{d,20}(\%)$
1	Robot Only	3.7	x
3	$\beta = 1$	10.0	12.7
3	$\beta = 0.5$	16.6	4.1
4	robot noise, $\bar{n}_r$	4.6	x
5	human noise, $\bar{n}_h$	18.3	x

output $\mathbf{q}_h$ eventually becomes small as the number of iterations increase. Therefore, with a lower level of assistance $\beta = 0.5$, the tracking error $\bar{e}_k$, defined in (54), converged to a higher value. This effect is seen by comparing the tracking error for the lower level of assistance $\beta = 0.5$, in Fig. 17 and Fig. 19, and for the higher level of assistance $\beta = 1$, in Fig. 13 and Fig. 15. In the later, the tracking error $\bar{e}_k$ converged closer to the robot noise $\bar{n}_r$, defined in (55).

Figure 20 compares the amount of effort contributed by the robot and human for both the co-tracking ($\beta = 0.5$) and skill-transfer ($\beta = 1$) cases. Effort, at each iteration step k, was quantified as

$$F_{i,k}(\%) = \max_{t \in [0, T_p]} \frac{\sqrt{q_{i,x,k}(t)^2 + q_{i,y,k}(t)^2}}{q_{d,max}} \times 100, \qquad (59)$$

where i is either r (for robot) or h (for human), $q_{i,x,k}$ and $q_{i,y,k}$ are the output components (e.g., either the robot or the human), and the size of the desired output $q_{d,max}$ is defined in (48). It can be seen in Fig. 20(a) that, as expected, during the skill-transfer (e.g, $\beta = 1$) the robot effort $F_{r,k}$ is initially zero and increases to 100%, while the human effort begins at 100% and decreases towards zero as the number of iteration steps k increase. In earlier iteration steps (e.g., $k = 2-3$), the robot provided over 100% of the desired output, since it was still learning the desired output. On the other hand, during the 50% co-tracking case with $\beta = 0.5$, as seen in Fig. 20(b), the robot effort $F_{r,k}$ begins at zero and increase to about 50%, while the human effort $F_{h,k}$ initially provides 100% and decreases to 50%.

4.4 Ability to capture human-response dynamics

The proposed method captured the impact of human-response dynamics during the human-robot collaboration. In order to analyze the influence of the human-response dynamics, the effect of the robot dynamics, e.g., in the x axis, $G_{R,x}(\omega)$ in (56) and shown in Fig. 11 was removed from the the learned models $\hat{G}_{x,k}$ (and similarly in $\hat{G}_{y,k}$) in (33) at each iteration step k by using the expression for $\overline{G}$ in (22), as

$$\tilde{G}_{x,k}(\omega) = \frac{\hat{G}_{x,k}(\omega)}{\rho(\omega) G_{R,x}(\omega)} \approx \left(1 - \frac{\beta}{1 + G_{H,x}(\omega)}\right). \qquad (60)$$

These normalized models $\tilde{G}_{x,k}$ and $\tilde{G}_{y,k}$ are shown in Fig. 21, for frequencies where the models had been updated at least once (e.g., see Lemma in Sect. 2.5), for both: (i) the skill-transfer case with level of assistance $\beta = 1$, and (ii) the co-tracking case with level of assistance $\beta = 0.5$. Note, that if the human-robot dynamics $\hat{G}_{x,k}$ did not depend on the human, then the phase in the normalized Fig. 21(b), would remain near zero degrees for all frequencies where the model was updated at some iteration step k. The results show that, in both cases, the phase in Fig. 21(b) for the x-axis, and Fig. 21(d) for the y-axis, deviated from zero degrees. Thus, these results indicate that the proposed method captures the impact of potentially-varying human-response dynamics $G_H(\omega)$ on the human-robot dynamics $\hat{G}_k$ during the iterative process.

5 Conclusion

This paper developed a data-driven iterative model inversion algorithm for human-robot collaborative output tracking when the task is unknown to the robot. Sufficient conditions for convergence of the algorithm were developed in the article. The proposed approach was verified using a robot-only experiment, which demonstrated that the proposed algorithm reduced the maximum error to 3.7%, where the closed-loop tracking noise was 4.6%. The algorithm also was tested during a human-robot collaborative experiment. Experimental results also showed that the level of assistance can be effectively modulated.

References

1. Bjoern Matthias, Soenke Kock, Henrik Jerregard, Mats Källman, and Ivan Lundberg. Safety of collaborative industrial robots: Certification possibilities for a collaborative assembly robot concept. In *Assembly and Manufacturing (ISAM), 2011 IEEE International Symposium on*, pages 1–6. IEEE, 2011.
2. Adriana Tapus, Maja J Mataric, and B Scasselati. Socially assistive robotics [grand challenges of robotics]. *Robotics & Automation Magazine, IEEE*, 14(1):35–42, 2007.
3. Marcus Mast, Michael Burmester, Katja Krüger, Sascha Fatikow, Georg Arbeiter, Birgit Graf, Gernot Kronreif, Lucia Pigini, David Facal, and Renxi Qiu. User-centered design of a dynamic-autonomy remote interaction concept for manipulation-capable robots to assist elderly people in the home. *Journal of Human-Robot Interaction*, 1(1):96–118, 2012.

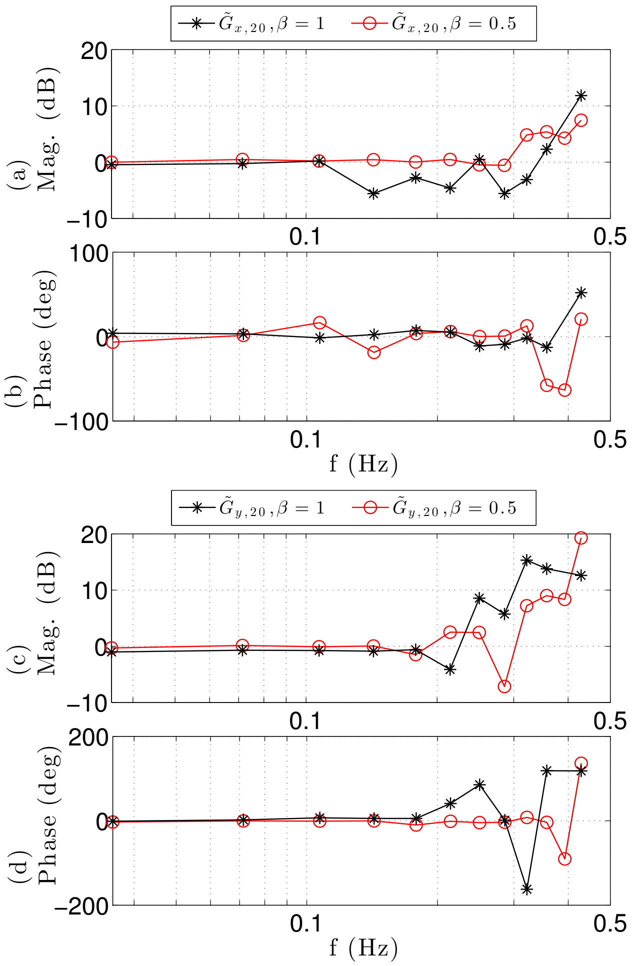

Fig. 21 Normalized models $\tilde{G}_{x,k}$ and $\tilde{G}_{y,k}$, defined in (60), showing the influence of human dynamics during human-robot collaboration with level of assistance $\beta = 1$ and $\beta = 0.5$: (a) magnitude of $\tilde{G}_{x,k}$ in decibels, (b) phase of $\tilde{G}_{x,k}$ in degrees, (c) magnitude of $\tilde{G}_{y,k}$ in decibels, (d) phase $\tilde{G}_{y,k}$ in degrees. Model data are only provided at frequencies where the model was updated at least once during the experiments (see Lemma in Sect. 2.5).

4. Neta Ezer, Arthur D Fisk, and Wendy A Rogers. More than a servant: Self-reported willingness of younger and older adults to having a robot perform interactive and critical tasks in the home. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 53, pages 136–140. SAGE Publications Sage CA: Los Angeles, CA, 2009.
5. Christian Martens, Oliver Prenzel, and Axel Gräser. *The rehabilitation robots FRIEND-I & II: Daily life independency through semi-autonomous task-execution.* Citeseer, 2007.
6. Tauseef Gulrez and Alessandro Tognetti. A Sensorized Garment Controlled Virtual Robotic Wheelchair. *Journal of Intelligent & Robotic Systems*, 74(3):847–868, June, 2014.
7. Jon Eriksson, Maja J Mataric, and C Winstein. Hands-off assistive robotics for post-stroke arm rehabilitation. In *Proc. IEEE International Conference on Rehabilitation Robotics (ICORR'05)*, pages 21–24, 2005.
8. Michael A Goodrich and Alan C Schultz. Human-robot interaction: a survey. *Foundations and trends in human-computer interaction*, 1(3):203–275, 2007.
9. Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, page 0278364913495721, 2013.
10. Duane T McRuer and Ezra S Krendel. The human operator as a servo system element. *Journal of the Franklin Institute*, 267(6):511–536, 1959.
11. Hyo-Sung Ahn, YangQuan Chen, and Kevin L Moore. Iterative learning control: brief survey and categorization. *IEEE Transactions on Systems Man and Cybernetics part C Applications and Reviews*, 37(6):1099, 2007.
12. Suguru Arimoto, Sadao Kawamura, and Fumio Miyazaki. Bettering operation of robots by learning. *Journal of Robotic systems*, 1(2):123–140, 1984.
13. Jur Van Den Berg, Stephen Miller, Daniel Duckworth, Humphrey Hu, Andrew Wan, Xiao-Yu Fu, Ken Goldberg, and Pieter Abbeel. Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 2074–2081. IEEE, 2010.
14. Jian Wang, Xin Xu, Daxue Liu, Zhenping Sun, and Qingyang Chen. Self-learning cruise control using kernel-based least squares policy iteration. *Control Systems Technology, IEEE Transactions on*, 22(3):1078–1087, 2014.
15. Rahul B Warrier and Santosh Devasia. Iterative learning from novice human demonstrations for output tracking. *IEEE Transactions on Human-Machine Systems*, 46(4):510–521, 2016.
16. Rahul B Warrier and Santosh Devasia. Inferring intent for novice human-in-the-loop iterative learning control. *IEEE Transactions on Control Systems Technology*, 2016.
17. Weihui Liu, Diansheng Chen, and Jochen Steil. Analytical Inverse Kinematics Solver for Anthropomorphic 7-DOF Redundant Manipulators with Human-Like Configuration Constraints. *Journal of Intelligent & Robotic Systems*, 86(1):63–79, April, 2017.
18. C. G. Atkeson and J. McIntyre. Robot trajectory learning through practice. In *IEEE Int. Conf. on Robotics and Automation*, pages 1737–1742, 1986.
19. M. Norrlof. An adaptive iterative learning control algorithm with experiments on an industrial robot. *IEEE Transactions on Robotics and Automation*, 18(2):245–251, Apr 2002.
20. X. Gao and S. Mishra. An iterative learning control algorithm for portability between trajectories. In *2014 American Control Conference, Portland, OR*, pages 3808–3813, June 4-6 June, 2014.
21. H.-S. Ahn, Y. Q. Chen, and K. L. Moore. Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(6):1099–1121, 2007.
22. J. Ghosh and B. Paden. Iterative learning control for nonlinear nonminimum phase plants. *ASME Journal of Dynamic Systems, Measurement, and Control*, 123:21–20, March,2001.
23. Szuchi Tien, Qingze Zou, and Santosh Devasia. Iterative control of dynamics-coupling-caused errors in piezoscanners during high-speed afm operation. *Control Systems Technology, IEEE Transactions on*, 13(6):921–931, 2005.
24. C. T. Freeman, P. L. Lewin, E. Rogers, D. H. Owens, and J. J. Hatonen. Discrete fourier transform based iterative

learning control design for linear plants with experimental verification. *ASME Journal of Dynamic Systems, Measurement, and Control*, 131(3):031006–1 – 031006–10, May,2009.

25. Kyong-Soo Kim and Qingze Zou. A modeling-free inversion-based iterative feedforward control for precision output tracking of linear time-invariant systems. *IEEE/ASME Transactions on Mechatronics*, 18(6):1767–1777, 2013.
26. Jonathan Realmuto, Rahul B Warrier, and Santosh Devasia. Iterative learning control for human-robot collaborative output tracking. In *Mechatronic and Embedded Systems and Applications (MESA), 2016 12th IEEE/ASME International Conference on*, pages 1–6. IEEE, 2016.
27. M Bruce Wiggin, Gregory S Sawicki, and Steven H Collins. An exoskeleton using controlled energy storage and release to aid ankle propulsion. In *Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on*, pages 1–5. IEEE, 2011.
28. Daniel Ragonesi, Sunil K Agrawal, Whitney Sample, and Tariq Rahman. Quantifying anti-gravity torques for the design of a powered exoskeleton. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 21(2):283–288, 2013.
29. JCK Lai, MP Schoen, A Perez Gracia, DS Naidu, and SW Leung. Prosthetic devices: challenges and implications of robotic implants and biological interfaces. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 221(2):173–183, 2007.
30. Claudia Perez-Maldonado, Anthony S Wexler, and Sanjay S Joshi. Two-dimensional cursor-to-target control from single muscle site semg signals. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(2):203–209, 2010.
31. Justin C Sanchez, Babak Mahmoudi, Jack DiGiovanna, and Jose C Principe. Exploiting co-adaptation for the design of symbiotic neuroprosthetic assistants. *Neural Networks*, 22(3):305–315, 2009.
32. Nicholas A Kirsch, Naji A Alibeji, and Nitin Sharma. Model predictive control-based dynamic control allocation in a hybrid neuroprosthesis. In *ASME 2014 Dynamic Systems and Control Conference*, pages V003T43A003–V003T43A003. American Society of Mechanical Engineers, 2014.
33. Bo Yu. *Interaction Dynamics in Oscillator and Human-in-the-loop Systems*. PhD thesis, The University of Michigan, 2014.
34. Stephen Butterworth. On the theory of filter amplifiers. *Wireless Engineer*, 7(6):536–541, 1930.
35. Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.
36. Stephen Timoshenko. *Strength of materials*. New York, 1930.