Dual-Stage Iterative Learning Control for MIMO Mismatched System With Application to Robots With Joint Elasticity

Wenjie Chen, *Member, IEEE*, and Masayoshi Tomizuka, *Fellow, IEEE*

***Abstract*— This paper discusses the tracking control problem for a class of multi-input–multi-output mismatched systems, where there are disturbances in different channels from the control input and the real-time feedback signal is not the output of interest. These mismatched issues make it difficult to achieve high tracking performance for the output of interest. To address this problem, two model-based iterative learning control (ILC) algorithms, namely reference ILC and torque ILC, are designed for different injection locations in the closed-loop system. An *ad hoc* hybrid scheme is proposed to make transitions between the two ILC stages for them to work properly at the same time. The application to both single-joint and multi-joint robots with joint elasticity are discussed with complete dynamic modeling and state estimation method for desired learning, as well as the extensive experimental validations on both systems.**

***Index Terms*— Elastic joint, industrial robot, iterative learning control (ILC), mismatched dynamics, multi-input, multi-output (MIMO) system.**

I. Introduction

IN INDUSTRIAL applications, the automated system (e.g., the robotic manipulator) is often required to repeatedly perform a single task under the same operating conditions. If the system repeatability is good, the trajectory tracking error will become repetitive from one run to another. In this case, the iterative learning control (ILC) scheme is well suited to compensate for the repetitive tracking error [1], [2].

ILC is a data-driven methodology that iteratively uses the data (e.g., error profile) from previous trails to update the system inputs for the next iteration. Normally, the ILC performs as an add-on feedforward controller in addition to the existing real-time feedback controller, to further enhance the performance over the standalone real-time feedback system. Many variations of the ILC scheme have been studied for various applications [1]. Most of them, however, are developed for the fundamental case where the system has direct measurement of the output of interest for real-time feedback and does not have disturbances in the channels different from the control input. Therefore, in mismatched systems where the above scenario does not hold, the standard ILC performance will be limited and in some cases the ILC convergence is hard to guarantee.

The mismatched systems discussed in this paper are common in practical applications, e.g., the industrial robots with indirect-drive joint mechanisms (joints with elasticity). The modeling and control of this kind of robots was discussed in [3], where the derived nonlinear dynamic model was shown to be globally linearizable and could be controlled by nonlinear state feedback based on a singular perturbation formulation of the model. Several other feedback control approaches, such as integrator backstepping [4], dynamic surface control [5], and adaptive robust control [6], have also been developed specifically to deal with the general mismatched systems. Some efforts have been devoted to migrate these ideas to the field of ILC to deal with the mismatched uncertainty iteratively while exploiting the noncausal repetitiveness. A dual-stage ILC approach was proposed in [7] to deal with robots with joint elasticity. Similar to backstepping, the real-time measured output (i.e., motor side state) is utilized in [7] as a hypothetical input to control the output of interest (i.e., load side state). As shown later in this paper, the convergence of this learning process may be adversely affected by the mismatched dynamics and thus the use of a high bandwidth Q filter to learn high-frequency error may compromise stability. Other studies such as [8] also reported the compromise on the tracking performance they had to make for a better learning convergence. This is especially the case when the system exhibits mismatched uncertainties. Regarding this stability issue under uncertainty, various robust approaches have been proposed [9], [10]. The resulting algorithms are usually nontrivial and the performance is normally compromised for a conservative robust controller. In [10], the plant resonances had to be suppressed by feedback compensation for the proposed method to improve the robustness to high-frequency modeling errors. In [11], a model-based ILC approach was developed for elastic robots to learn the error component beyond the first resonant frequency. However, this approach requires an accurate piecewise-linear model to be identified and interpolated for each trajectory in advance, which limits its application.

In our recent work [12], [13], a hybrid dual-stage model-based ILC approach was proposed for a class of multi-input, multi-output (MIMO) mismatched systems. The dual-stage

Manuscript received November 11, 2012; revised July 12, 2013; accepted August 16, 2013. Manuscript received in final form August 22, 2013. This work was supported by FANUC Corporation, Japan. Real-time control hardware and software for the single-joint robotic testbed were provided by National Instruments, Inc. Recommended by Associate Editor F. Chowdhury.

The authors are with the Department of Mechanical Engineering, University of California, Berkeley, CA 94720 USA (e-mail: wjchen@berkeley.edu; tomizuka@me.berkeley.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2013.2279652

ILC aims to push the learning algorithm to a higher bandwidth while maintaining the fast model-based convergence rate. In this paper, the systematic formulation and analysis of the algorithm design, and more extensive applications to both single-joint indirect drives and multi-joint robots with joint elasticity will be presented, compared with our prior work, which was more focused on the specific design for that particular application presented therein.

This paper is organized as follows. The system model and the basic controller structure are introduced first. Then two ILC schemes are designed independently followed by an *ad hoc* hybrid scheme to enable the two ILC schemes to execute simultaneously. The application to a single-joint indirect drive system is presented next to validate the effectiveness of the proposed scheme. The parametric uncertainty and mismatched dynamics such as various disturbances at different locations of the system will be addressed. The proposed scheme is also applied to multi-joint robotic systems. The robot dynamic model formulation and the load side state estimation algorithm will be presented as well. Then the experimental validation on a six-joint industrial robot will be provided. The conclusions of this paper are given at last.

II. System Overview

A. System Model

Consider a MIMO mismatched system in the following discrete time state-space form:

$$x(j+1) = Ax(j) + B_u u(j) + B_d d(j) \tag{1}$$

$$\begin{aligned} y(j) &= \left[q_m^{\mathrm{T}}(j),\, q_\ell^{\mathrm{T}}(j)\right]^{\mathrm{T}} \\ &= Cx(j) + D_u u(j) + D_d d(j) \end{aligned} \tag{2}$$

where j is the time step index, $x \in \mathbb{R}^{n_x}$ is the system state, $u \in \mathbb{R}^{n_u}$ is the control input, $d \in \mathbb{R}^{n_d}$ is the lumped disturbance, and $q_m \in \mathbb{R}^{n_m}$ and $q_\ell \in \mathbb{R}^{n_\ell}$ are the two outputs of the plant.[1] d is regarded as the mismatched uncertainty/disturbance if it (or part of it) is applied to different channels from the control input u (i.e., $B_u \neq \alpha B_d, \forall \alpha \in \mathbb{R}$). Note that, d can include the system nonlinearity for modeling of a nonlinear system. Another mismatched assumption is that, only part of the outputs (i.e., q_m) is measured for real-time feedback, even if the output of interest may be q_ℓ. However, the measurement/estimate of q_ℓ may be obtained offline for iteration-based applications. These assumptions are reasonable as seen from the two applications later. Furthermore, besides the unknown mismatched dynamics, it is assumed that parametric uncertainties exist in the available nominal model.

This system can be reformulated as

$$q_m(j) = P_{\mathrm{mu}}(z)u(j) + P_{\mathrm{md}}(z)d(j) \tag{3}$$

$$q_\ell(j) = P_{\ell u}(z)u(j) + P_{\ell d}(z)d(j) \tag{4}$$

where P_{mu}, P_{md}, $P_{\ell u}$, and $P_{\ell d}$ are the transfer functions from u or d to the corresponding output. For simplicity, the following control scheme is formulated for the case where P_{mu} and $P_{\ell u}$ are diagonal matrices. However, it may be possible to extend this paper to a more general case by considering the plant inversion and commutative multiplication for the non-diagonal matrices. Alternatively, we can also apply this diagonal formulation to general MIMO systems by model decoupling. As shown later in the multi-joint robot case (Section V), which is a highly coupled nonlinear MIMO system, the dynamic model is decoupled by grouping the coupling terms and the nonlinear terms into the lumped disturbance d, thus resulting in the diagonal formulation of (3) and (4).

[1]The utilization of the subscripts m (motor side) and ℓ (load side) is to be consistent with the later application studies.

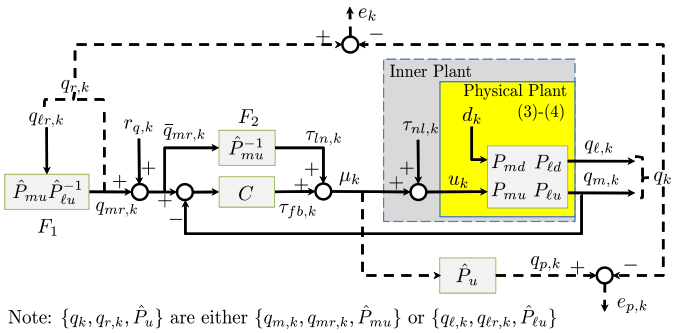


Fig. 1. Control structure with reference and torque updates. Solid lines: real-time signals. Dashed lines: signals for offline use. Subscript k: iteration index. The reference input is $q_{r,k}$ and the corresponding plant output to be controlled is q_k. e_k and $e_{p,k}$ are the tracking error and the model following error used for iterative learning, which generate the reference and feedforward torque updates, $r_{q,k}$ and $\tau_{\mathrm{nl},k}$, respectively.

B. Basic Controller Structure

Fig. 1 shows the control structure for this mismatched system, where the subscript k is the iteration index. The proposed control structure has two nested loops. The inner loop (i.e., feedback controller C and feedforward torque signal consisting of the linear component τ_{ln} from F_2 and the nonlinear component τ_{nl}) uses the control input u to the physical plant to achieve tracking of the measurable output q_m. The desired tracking of the output of interest q_ℓ is addressed in the outer loop (i.e., feedforward reference controller F_1 and feedforward reference signal r_q), which generates the reference of the measurable output q_{mr} as the control to the inner loop. Here, C can be any linear feedback controller such as a decoupled PID controller to stabilize the system. The feedforward controllers, F_1 and F_2, are designed using the nominal inverse model as

$$q_{\mathrm{mr},k}(j) = \hat{P}_{\mathrm{mu}}(z)\hat{P}_{\ell u}^{-1}(z)q_{\ell r,k}(j) \triangleq F_1(z)q_{\ell r,k}(j) \tag{5}$$

$$\tau_{\mathrm{ln},k}(j) = \hat{P}_{\mathrm{mu}}^{-1}(z)\left[q_{\mathrm{mr},k}(j) + r_{q,k}(j)\right] \triangleq F_2(z)\bar{q}_{\mathrm{mr},k}(j) \tag{6}$$

where $\hat{\bullet}$ is the nominal model representation of $\bullet$, $q_{\ell r,k}$ is the plant reference output for $q_{\ell,k}$, and $r_{q,k}$ and $\tau_{\mathrm{nl},k}$ are used, respectively, as the additional reference and feedforward torque updates generated iteratively by the dual-stage ILC algorithm designed later.

III. Dual-Stage ILC Scheme

A. ILC Basics

Some basics of the general ILC scheme are reviewed first, which will be utilized in the subsequent ILC scheme design.

Consider the MIMO system with the error dynamics and the ILC law in the following form:

$$\bar{e}_k(j) = -P_{\text{eu}}(z)\bar{u}_k(j) + \bar{r}(j) \tag{7}$$

$$\bar{u}_{k+1}(j) = Q(z)\left[\bar{u}_k(j) + L(z)\bar{e}_k(j)\right] \tag{8}$$

where $\bar{e}$ is the error that the ILC scheme aims to reduce, $\bar{r}$ is the lumped repetitive reference and/or disturbance input to the system, and $\bar{u}$ is the control input updated iteratively by the ILC scheme using the filters $L(z)$ and $Q(z)$. Similar to [1] and [14], the following convergence property holds.

Theorem 1: The ILC system (7), (8) is monotonically and exponentially convergent in the sense that $\|\bar{u}_k - \bar{u}_\infty\|_p \to 0$ and $\|\bar{e}_k - \bar{e}_\infty\|_p \to 0$ as $k \to \infty$, if

$$\beta = \|Q(z)\left[I - L(z)P_{\text{eu}}(z)\right]\|_p < 1 \tag{9}$$

where β is the rate of convergence, I is the identity matrix with appropriate dimension, the p-norm $\|\bullet\|_p = (\Sigma_i |\bullet_i|^p)^{1/p}$, and

$$\bar{u}_\infty(j) = \left[I - Q(z) + Q(z)L(z)P_{\text{eu}}(z)\right]^{-1} Q(z)L(z)\bar{r}(j) \tag{10}$$

$$\bar{e}_\infty(j) = \left[I - Q(z) + Q(z)L(z)P_{\text{eu}}(z)\right]^{-1}\left[I - Q(z)\right]\bar{r}(j). \tag{11}$$

Proof: First, with (7) and (8), it is easy to see that

$$\bar{u}_{k+1}(j) = Q(z)\left[I - L(z)P_{\text{eu}}(z)\right]\bar{u}_k(j) + Q(z)L(z)\bar{r}(j)$$

which yields

$$\begin{aligned}\|\bar{u}_{k+1} - \bar{u}_\infty\|_p &= \|Q(z)\left[I - L(z)P_{\text{eu}}(z)\right](\bar{u}_k - \bar{u}_\infty)\|_p \\ &\le \|Q(z)\left[I - L(z)P_{\text{eu}}(z)\right]\|_p \, \|\bar{u}_k - \bar{u}_\infty\|_p .\end{aligned}$$

Therefore, if $\|Q(z)\left[I - L(z)P_{\text{eu}}(z)\right]\|_p < 1$, $\|\bar{u}_k - \bar{u}_\infty\|_p \to 0$ as $k \to \infty$. With (7), similar conclusion can be drawn for the convergence of $\bar{e}_k$. Note that the inverse $[I - Q(z) + Q(z)L(z)P_{\text{eu}}(z)]^{-1}$ exists because $\|Q(z)[I - L(z)P_{\text{eu}}(z)]\|_p < 1$. ∎

Equation (11) shows that the steady-state error $\bar{e}_\infty$ vanishes with complete learning (i.e., $Q(z) = I$), which means the effects on $\bar{e}_\infty$ from the repetitive input $\bar{r}$ will be fully compensated. The ILC law (8) implies that the Q filter can also be used to shape the learning ability in the frequency domain. To achieve better performance, it is desired to push the bandwidth of $Q(z)$ to be as high as possible. Equation (9), however, shows that the bandwidth of $Q(z)$ may have to be compromised to ensure monotonic convergence and to avoid poor transients in the learning process. In practice, a low-pass filter $Q(z)$ is typically employed to prevent the effects of high-frequency model uncertainties from entering the learning process [1]. Also, $Q(z)$ should be unity gain at low frequencies where complete learning is preferred to achieve zero steady-state error. Because the ILC scheme is an offline iteration-based method, acausal filtering can be utilized to obtain a zero-phase learning response.

Given a fixed Q filter, the optimal learning filter to achieve the fastest convergence is obtained as

$$L^*(z) = \arg\min_{L(z)} \|Q(z)\left[I - L(z)P_{\text{eu}}(z)\right]\|_p . \tag{12}$$

This leads to the plant inversion choice, i.e., $L^*(z) = P_{\text{eu}}^{-1}(z)$. This model matching problem can be solved with many optimization techniques, such as the H_∞ synthesis [9], if the model uncertainty bound is known. The designed $Q(z)$ and $L(z)$ need to be validated using (9) with the knowledge of system model uncertainty. As the actual plant dynamics P_{eu} is often not exactly known, a suboptimal learning filter is chosen in this paper simply as the nominal model inverse $L^*(z) = \hat{P}_{\text{eu}}^{-1}(z)$.

B. ILC With Reference Update

Denote the sensitivity function of the closed-loop system in Fig. 1 as $S_p(z) = [I + C(z)P_{\text{mu}}(z)]^{-1}$. From (5), $q_{\text{mr},k}$ is related to $q_{r,k}$ (i.e., $q_{\text{mr},k}$ or $q_{\ell r,k}$) as follows:

$$q_{\text{mr},k} = \hat{P}_{\text{mu}}\hat{P}_u^{-1}q_{r,k} \tag{13}$$

where P_u can be either P_{mu} or $P_{\ell u}$ depending on the choice of $q_{r,k}$. The time index j for all signals and the discrete time operator z for all transfer functions are omitted hereafter for simplicity.

To reduce the tracking error $e_k = q_{r,k} - q_k$, an ILC scheme with reference update can be applied according to the following lemma.

Lemma 1: If the reference trajectory $q_{r,k}$, the feedforward torque update[2] $\tau_{\text{nl},k}$, and the disturbance d_k are repetitive[3] for each iteration, the tracking error e_k will be monotonically decreasing in the iteration domain by the ILC scheme

$$r_{q,k+1} = Q_r(r_{q,k} + L_r^* e_k) \tag{14}$$

$$L_r^* = \hat{P}_u^{-1}\hat{P}_{\text{mu}} \tag{15}$$

as long as the convergence rate β_r^* satisfies

$$\beta_r^* = \left\| Q_r(I - \hat{P}_u^{-1}P_u S_p \hat{S}_p^{-1})\right\|_\infty < 1. \tag{16}$$

Proof: The system output q_k (i.e., $q_{m,k}$ or $q_{\ell,k}$) can be derived as

$$\begin{aligned} q_k &= P_u u_k + P_d d_k \\ &= P_u S_p\left[(C + \hat{P}_{\text{mu}}^{-1})(q_{\text{mr},k} + r_{q,k}) + \tau_{\text{nl},k} - C P_{\text{md}} d_k\right] + P_d d_k \\ &= \hat{P}_{\text{mu}}^{-1} P_u S_p \hat{S}_p^{-1} r_{q,k} + P_u S_p \\ &\quad \cdot \left(\hat{P}_u^{-1}\hat{S}_p^{-1} q_{r,k} + \tau_{\text{nl},k} - C P_{\text{md}} d_k\right) + P_d d_k \end{aligned} \tag{17}$$

where we have noted that C, P_{mu}, $P_{\ell u}$, and their nominal models ($\hat{P}_{\text{mu}}$ and $\hat{P}_{\ell u}$) are diagonal. Similarly to P_u, P_d can be either P_{md} or $P_{\ell d}$ depending on the choice of $q_{r,k}$. The

[2]This implies that the ILC with torque update (see the next section) is inactivated. If the torque update is also activated, the repetitiveness assumption may be invalid and we will need to use the hybrid scheme introduced later.

[3]The disturbance considered here includes external disturbances (such as the friction force or the physical interaction with the environment) and fictitious disturbances (such as model inaccuracy or unmodeled nonlinearity). These components are normally repetitive if the reference trajectory is repetitive, which is often the case in industrial applications where the tasks are performed repeatedly. The initial state change or random noises could also be treated as part of the disturbance. These parts are generally not repetitive and thus cannot be compensated by the proposed ILC method. However, they will not affect the performance much, since in most applications the magnitudes of the initial state change and random noises are negligible compared with the magnitudes of other major repetitive disturbance terms described above.

corresponding tracking error e_k is

$$\begin{aligned} e_k &= q_{r,k} - q_k \\ &= -\hat{P}_{\text{mu}}^{-1} P_u S_p \hat{S}_p^{-1} r_{q,k} + (I - P_u S_p \hat{P}_u^{-1} \hat{S}_p^{-1}) q_{r,k} \\ &\quad - P_u S_p \tau_{\text{nl},k} + (P_u S_p C P_{\text{md}} - P_d) d_k \\ &\triangleq -P_{eu,r} r_{q,k} + \bar{r}_{r,k}. \end{aligned} \tag{18}$$

Therefore, the tracking performance of the next iteration can be improved by the ILC scheme with reference update (14) (namely reference ILC (L) for $e_k = e_{\ell,k}$ or reference ILC (M) for $e_k = e_{m,k}$), if the reference trajectory $q_{r,k}$, the feedforward torque update $\tau_{\text{nl},k}$, and the disturbance d_k are repetitive for each iteration. From Theorem 1, the monotonic convergence of this ILC scheme (14) can be guaranteed if $\beta_r = \|Q_r(1 - L_r \hat{P}_{\text{mu}}^{-1} P_u S_p \hat{S}_p^{-1})\|_\infty < 1$. With the inversion of the nominal model in (18) (i.e., $\hat{P}_{eu,r}^{-1} = \hat{P}_u^{-1} \hat{P}_{\text{mu}}$), the suboptimal learning filter and the convergence rate are obtained as (15) and (16), respectively. ■

Remark 1: With complete learning (i.e., $Q_r = I$), the tracking error e_∞ vanishes when the convergence condition (16) is met.

Remark 2: It is seen in (16) that, the model uncertainty will greatly affect the convergence rate. To achieve fast convergence rate without compromising the bandwidth of Q_r, it is desired to reduce the model uncertainties, i.e., $\hat{P}_u \approx P_u$ and $\hat{S}_p \approx S_p$ are desired. Since the sensitivity function S_p is robust within the bandwidth, this objective can be achieved by obtaining a nominal model $\hat{P}_u$ more accurately representing the actual physical plant. This will require a significant effort for system identification and the resulting nominal model may be high order or highly nonlinear, which is nontrivial for control use.

In contrast, the same objective can be achieved by making the inner plant (grey shaded area with dashed outlines in Fig. 1) behave as the chosen nominal model $\hat{P}_u$. In the following section, an ILC scheme with torque update is introduced to accomplish this, i.e., making $q_k \to \hat{P}_u \mu_k$, where $\mu_k = \tau_{\text{In},k} + \tau_{\text{fb},k}$ is the torque input to the inner plant.

C. ILC With Torque Update

Define $e_{p,k}$ as the model following error between the nominal plant output ($q_{p,k} \triangleq \hat{P}_u \mu_k$) and the actual plant output q_k ($q_{\ell,k}$ or $q_{m,k}$), i.e., $e_{p,k} = \hat{P}_u \mu_k - q_k \triangleq q_{p,k} - q_k$, where $\hat{P}_u$ is $\hat{P}_{\ell u}$ or $\hat{P}_{\text{mu}}$ to match with the choice of q_k. The ILC scheme to reduce this model following error $e_{p,k}$ can be then formulated as

$$\tau_{\text{nl},k+1} = Q_u(\tau_{\text{nl},k} + L_u e_{p,k}). \tag{19}$$

The corresponding ILC is termed as torque ILC (L) for $e_{p,k} = \hat{P}_{\ell u} \mu_k - q_{\ell,k}$ or torque ILC (M) for $e_{p,k} = \hat{P}_{\text{mu}} \mu_k - q_{m,k}$.

Remark 3: As shown in Fig. 1, $\tau_{\text{nl},k}$ is injected inside the inner plant to cancel out the effects of model uncertainty $\Delta P \triangleq P_u - \hat{P}_u$ and mismatched disturbance d_k on q_k. The ideal $\tau_{\text{nl},k}^*$ to achieve this objective can be derived as

$$\begin{aligned} \hat{P}_u \mu_k &= P_u(\mu_k + \tau_{\text{nl},k}^*) + P_d d_k \\ \Rightarrow \tau_{\text{nl},k}^* &= -P_u^{-1}(\Delta P \mu_k + P_d d_k). \end{aligned} \tag{20}$$

In the mismatched systems, the two objectives, following $\hat{P}_{\text{mu}}$ [i.e., torque ILC (M)] and following $\hat{P}_{\ell u}$ [i.e., torque ILC (L)], cannot be attained simultaneously (i.e., $\tau_{\text{nl,mk}}^* \neq \tau_{\text{nl},\ell\text{k}}^*$), since $P_{\text{mu}}^{-1} P_{\text{md}} \neq P_{\ell u}^{-1} P_{\ell d}$ and $P_{\text{mu}}^{-1} \Delta P_m \neq P_{\ell u}^{-1} \Delta P_\ell$ in general. Thus, at this stage in view of (16), it is desired to select the nominal model to match with the chosen one in the reference ILC stage.

1) Convergence of Model Following Error:

Lemma 2: If the reference trajectory $q_{\text{mr},k}$, the reference update[4] $r_{q,k}$, and the disturbance d_k remain the same for each iteration, the model following error $e_{p,k}$ will be monotonically decreasing over iteration by the torque ILC scheme (19) if

$$L_u = L_u^* = \hat{P}_u^{-1} \tag{21}$$

$$\beta_u^* = \left\| Q_u \left(I - P_u \hat{P}_u^{-1} \right) S_p \right\|_\infty < 1. \tag{22}$$

Proof: The input–output equation of the nominal plant can be derived as

$$\begin{aligned} q_{p,k} &\triangleq \hat{P}_u \mu_k \\ &= \hat{P}_u S_p \big[(C + \hat{P}_{\text{mu}}^{-1})(q_{\text{mr},k} + r_{q,k}) \\ &\qquad - C P_{\text{mu}} \tau_{\text{nl},k} - C P_{\text{md}} d_k \big]. \end{aligned} \tag{23}$$

Then from (17) and (23), the model following error $e_{p,k}$ becomes:

$$\begin{aligned} e_{p,k} &= q_{p,k} - q_k \\ &= -T_u S_p \tau_{\text{nl},k} - \Delta P S_p (C + \hat{P}_{\text{mu}}^{-1}) \\ &\quad \cdot (q_{\text{mr},k} + r_{q,k}) + (\Delta P S_p C P_{\text{md}} - P_d) d_k \\ &\triangleq -P_{eu,u} \tau_{\text{nl},k} + \bar{r}_{u,k} \end{aligned} \tag{24}$$

where $T_u = \hat{P}_u C P_{\text{mu}} + P_u$.

Therefore, if the reference trajectory $q_{\text{mr},k}$, the reference update $r_{q,k}$, and the disturbance d_k remain the same for each iteration, by Theorem 1, the torque ILC scheme (19) will be monotonically converging if $\beta_u = \| Q_u \left(I - L_u T_u S_p \right) \|_\infty < 1$. Using the nominal plant inversion $\hat{P}_{eu,u}^{-1} = \hat{S}_p^{-1} \hat{T}_u^{-1} = \hat{P}_u^{-1}$, the suboptimal choice of L_u and the convergence rate β_u become (21) and (22), respectively. ■

Remark 4: With complete learning (i.e., $Q_u = I$), the inner plant behaves like the nominal model as the model following error $e_{p,\infty} \to 0$ when the convergence condition (22) is met.

2) Convergence of Tracking Error:

Lemma 3: If $q_{r,k}$, $r_{q,k}$, and d_k do not vary from one iteration to another, the tracking error e_k will converge monotonically with the rate of $\beta_e \leq \| P_u T_u^{-1} \|_\infty \beta_u$ as long as the model following error $e_{p,k}$ converges and $P_u T_u^{-1}$ is bounded-input bounded-output (BIBO) stable.

Proof: Using (18) and (24), the tracking error e_k can be derived as

$$\begin{aligned} e_k = P_u T_u^{-1} \Big[& e_{p,k} + \hat{P}_{\text{mu}} C \left(P_u^{-1} P_{\text{mu}} \hat{P}_u \hat{P}_{\text{mu}}^{-1} - I \right) q_{r,k} \\ & - \left(\hat{P}_{\text{mu}}^{-1} + C \right) \hat{P}_u r_{q,k} + C \hat{P}_u \left(P_{\text{md}} - P_d P_{\text{mu}} P_u^{-1} \right) d_k \Big]. \end{aligned} \tag{25}$$

Thus, the conclusion in the lemma is obtained. ■

[4]This implies that the ILC with reference update is inactivated. If the reference update is also activated, the repetitiveness assumption may be invalid and we will need to use the hybrid scheme introduced later.

Remark 5: For torque ILC (M) (i.e., $P_u = P_{\text{mu}}$ and $P_d = P_{\text{md}}$), $P_u^{-1}P_{\text{mu}}\hat{P}_u\hat{P}_{\text{mu}}^{-1} - I = 0$, and $P_{\text{md}} - P_d P_{\text{mu}} P_u^{-1} = 0$, which further reduces (25) to

$$e_k = P_u T_u^{-1}\big[e_{p,k} - (\hat{P}_{\text{mu}}^{-1} + C)\hat{P}_u r_{q,k}\big]. \tag{26}$$

Thus if $r_{q,k} = 0$ (i.e., the reference ILC is not activated at all), $e_k \to 0$ as $e_{p,k}$ vanishes.

Remark 6: For torque ILC (L) (i.e., $P_u = P_{\ell u}$ and $P_d = P_{\ell d}$), even if $r_{q,k} = 0$ and $e_{p,k}$ vanishes, $e_k \to 0$ is not necessarily true due to the mismatched behavior. The remaining tracking error e_∞ is

$$\begin{aligned} e_\infty &= P_u T_u^{-1}\Big[\hat{P}_{\text{mu}} C\big(P_u^{-1}P_{\text{mu}}\hat{P}_u\hat{P}_{\text{mu}}^{-1} - I\big)q_{r,\infty} - (\hat{P}_{\text{mu}}^{-1} + C) \\ &\qquad \times \hat{P}_u r_{q,\infty} + C\hat{P}_u(P_{\text{md}} - P_d P_{\text{mu}} P_u^{-1})d_\infty\Big] \\ &\triangleq -P_{eu,ur} r_{q,\infty} + \bar{r}_{ur,\infty} \end{aligned} \tag{27}$$

which can be further eliminated through the reference ILC using $L_r^* = \hat{P}_{eu,ur}^{-1} = \hat{P}_u^{-1}\hat{P}_{\text{mu}}$, and this matches with (15).

D. Hybrid Scheme With Dual-Stage ILC

In general, for the closed-loop system with a satisfactory feedback controller, the sensitivity function S_p will behave as a high-pass filter to mitigate the low-frequency error. Therefore, in the convergence condition (22), the low-frequency model uncertainty is greatly suppressed by S_p. This allows Q_u to have higher bandwidth without worrying about the low-frequency uncertainty. Then with the effects of the torque ILC, the inner plant will behave like the nominal model (i.e., $q_k \to \hat{P}_u\mu_k$) up to the bandwidth of Q_u. Within this frequency range, the convergence condition of the reference ILC (16) is simplified to

$$\beta_r \approx \left\| Q_r\left(I - S_p\hat{S}_p^{-1}\right)\right\|_\infty < 1 \tag{28}$$

which allows to push Q_r to a higher bandwidth for better tracking performance.

Note that the repetitive assumption is used in the derivation of the aforementioned two ILC schemes. When these two ILC schemes are activated simultaneously, the repetitive assumption will be no longer valid (i.e., $r_{q,k}$ and $\tau_{\text{nl},k}$ are not repetitive from one iteration to another). This may introduce the adverse interference of the two ILC stages. Therefore, an *ad hoc* hybrid scheme is designed to address this problem. Specifically, an iteration-varying gain is applied to each ILC stage as follows:

$$\tau_{\text{nl},k+1} = Q_u\left(\tau_{\text{nl},k} + \gamma_{u,k}L_u e_{p,k}\right) \tag{29}$$

$$r_{q,k+1} = Q_r(r_{q,k} + \gamma_{r,k}L_r e_k). \tag{30}$$

where the two gains $\gamma_{u,k}$ and $\gamma_{r,k}$ can be tuned by trial and error. The basic idea behind is that the torque ILC needs to take more effect whenever the model following error becomes larger in the previous iteration (e.g., $\|e_{p,k}^i\|_2/\|e_{p,1}^i\|_2$ increases, or $\|e_{p,k}^i\|_2/\|e_{p,k-1}^i\|_2$ is greater than 1, where $\|\bullet^i\|_p$ is the p-norm of the i-th component of $\bullet(j)$ along the time index j). For the better performance of torque ILC, the effect of the reference ILC is accordingly attenuated with a decreased $\gamma_{r,k}$. In contrast, once the model following error is sufficiently small (i.e., the inner plant behaves as the nominal model) or becomes stable from the previous iteration, the torque ILC becomes unnecessary and the reference ILC can be fully activated.

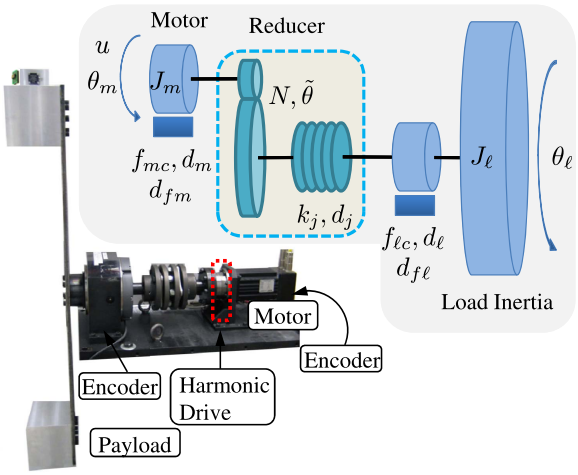


Fig. 2. Single-joint indirect drive system setup.

Remark 7: The proposed hybrid scheme aims to deal with the mismatched dynamics by improving the performance bandwidth of the ILC without compromising the stability. Equation (26) indicates that, for the application of tracking q_m, the reference ILC is not necessary and the torque ILC (M) with $\hat{P}_u = \hat{P}_{\text{mu}}$ will be sufficient. To track q_ℓ, however, the aforementioned hybrid dual-stage ILC scheme with $\hat{P}_u = \hat{P}_{\ell u}$ will be necessary.

Remark 8: It is understood that the nominal models used in two ILC stages should match with each other due to the mismatched dynamics. This means that these two stages need to be both load side learning or both motor side learning, but not learning on the two sides together, since the nominal behaviors of load side and motor side cannot be achieved simultaneously due to the mismatched dynamics. This will be demonstrated in the following application study, where the algorithm validation will focus on the case of tracking q_ℓ to test the effectiveness of the hybrid dual-stage ILC scheme.

Remark 9: As mentioned above, the ILC performs as an add-on feedforward controller in addition to the existing real-time feedback controller, which can be designed in many forms including adaptive and/or robust controllers. While the ILC provides the superior performance to deal with the predominant repetitive disturbance in the repetitive applications, the feedback controller is also necessary to deal with unexpected disturbances.

IV. Application to Single-Joint Indirect Drive Systems

The mismatched systems described in this paper are common in practice. One simple example is the single-joint indirect drive system shown in Fig. 2, where the motor is used to indirectly control the motion of the load subject to mismatched disturbances/uncertainties as stated later.

A. Dynamic Model

As shown by the schematic diagram in Fig. 2, the subscripts m and ℓ denote the motor side and the load side quantities,

respectively. θ represents the angular position and J is the moment of inertia. u is the motor torque input. d_m and d_ℓ represent the viscous damping coefficients at the motor side and the load side, respectively. k_j and d_j are the stiffness and the damping coefficients of the reducer. The gear ratio of the reducer is denoted by N. $f_{\rm mc}$ and $f_{\ell c}$ represent the nonlinear Coulomb frictions at the motor side and the load side, respectively. $d_{\rm fm}$ and $d_{f\ell}$ represent the additional repetitive disturbances at the motor side and the load side, respectively. $\tilde{\theta}$ is the transmission error of the harmonic drive, which is defined as the deviation between the expected reducer output position and the actual reducer output position. It can be approximated with a simple sinusoid as $\tilde{\theta} = A\sin(2\dot{\theta}_m t + \phi)$, where A is the amplitude of the transmission error, ϕ is the phase, and the frequency is two times the motor side velocity [15].

The dynamic model for this setup can be formulated as

$$J_m\ddot{\theta}_m + d_m\dot{\theta}_m = -\frac{1}{N}\left[k_j\left(\frac{\theta_m}{N} - \theta_\ell\right) + d_j\left(\frac{\dot{\theta}_m}{N} - \dot{\theta}_\ell\right)\right] + u + d_1 \quad (31)$$

$$J_\ell\ddot{\theta}_\ell + d_\ell\dot{\theta}_\ell = k_j\left(\frac{\theta_m}{N} - \theta_\ell\right) + d_j\left(\frac{\dot{\theta}_m}{N} - \dot{\theta}_\ell\right) + d_2 \quad (32)$$

where

$$d_1 = d_{\rm fm} - f_{\rm mc}\,\mathrm{sgn}(\dot{\theta}_m) + \frac{1}{N}\left(k_j\tilde{\theta} + d_j\dot{\tilde{\theta}}\right)$$

$$d_2 = d_{f\ell} - f_{\ell c}\,\mathrm{sgn}(\dot{\theta}_\ell) - \left(k_j\tilde{\theta} + d_j\dot{\tilde{\theta}}\right).$$

Therefore, the above indirect drive model can be considered as a mismatched system described in (3) and (4) with the disturbance $d = [d_1\ d_2]^{\rm T}$. The two outputs q_m and q_ℓ are the motor side position θ_m and the load side position θ_ℓ, respectively. Note that d_1 and d_2 are state-dependent disturbance terms, and thus d is repetitive if q_m and q_ℓ are repetitive. If the stability of the feedback system is preserved, the goal of the feedforward control is to cancel d_1 and d_2 that would correspond to the reference trajectories. The continuous time transfer functions from the inputs to the outputs become

$$P_{\rm mu}(s) = \frac{J_\ell s^2 + (d_j + d_\ell)s + k_j}{J_m J_\ell s^4 + J_d s^3 + J_k s^2 + k_j(d_m + d_\ell/N^2)s}$$

$$P_{\ell u}(s) = \frac{d_j s + k_j}{N\left[J_m J_\ell s^4 + J_d s^3 + J_k s^2 + k_j(d_m + d_\ell/N^2)s\right]}$$

$$P_{md_2}(s) = \frac{d_j s + k_j}{N\left[J_m J_\ell s^4 + J_d s^3 + J_k s^2 + k_j(d_m + d_\ell/N^2)s\right]}$$

$$P_{\ell d_2}(s) = \frac{J_m s^2 + (d_j/N^2 + d_m)s + k_j/N^2}{J_m J_\ell s^4 + J_d s^3 + J_k s^2 + k_j(d_m + d_\ell/N^2)s}$$

$$P_{\rm md}(s) = \left[P_{\rm mu}(s)\ P_{md_2}(s)\right],\ P_{\ell d}(s) = \left[P_{\ell u}(s)\ P_{\ell d_2}(s)\right]$$

where

$$J_d = J_m(d_j + d_\ell) + J_\ell\left(\frac{d_j}{N^2} + d_m\right)$$

$$J_k = J_m k_j + \frac{J_\ell k_j}{N^2} + (d_j + d_\ell)d_m + \frac{d_j d_\ell}{N^2}.$$

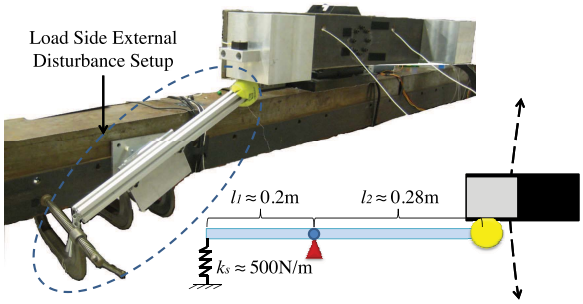


Fig. 3. Load side disturbance setup for a single-joint system.

B. Experimental Setup and System Disturbances

The proposed method is validated on a typical single-joint indirect drive robot testbed shown in Fig. 2. This experimental setup consists of: 1) a servo motor with a 20 000 counts/revolution encoder; 2) a harmonic drive with a 80:1 gear ratio; 3) a load side 144 000 counts/revolution encoder; and 4) a payload. The antiresonant and resonant frequencies of the setup are approximately 11 and 19 Hz. It is assumed that the load side encoder measurement is only available for iteration-based offline use rather than for real-time feedback use. Finally, the algorithms are implemented using a 1 kHz sampling rate in a LabVIEW real-time target installed with LabVIEW Real-Time and FPGA modules.

In this setup, the identified Coulomb friction combined at the motor side (i.e., $f_{\rm mc} + f_{\ell c}/N^2$) is about 0.1004 N·m. A fictitious torque is added in the motor torque command to simulate the external disturbance $d_{\rm fm}$ at the motor side. In the following experiments, $d_{\rm fm}$ is set as a 1 Hz sinusoid starting from 3 s with an amplitude of 0.2 N·m, i.e., $d_{\rm fm} = 0.2\sin(2\pi(t-3))h(t-3)$ N·m, where $h(t-3)$ is a Heaviside step function. The repetitive external disturbance $d_{f\ell}$ at the load side is generated using the setup shown in Fig. 3. It is designed to have the extension springs apply a maximum disturbance of approximately 20 N·m at the load side when the payload hits the ball and continues rotating for about 14°.

The motor side feedback controller C designed for this setup has a resonant frequency at about 11 Hz for the velocity loop. This corresponds to about 1 rad/s at the load side. Therefore, to amplify the transmission error effects, the load side reference trajectory is designed to have a speed of 0.5 rad/s for most time so that the resulting transmission error frequency will coincide with the resonant frequency of the velocity loop. The resulting trajectory is shown in Fig. 4, which is designed as a fourth-order time optimal trajectory suggested in [16].

The effects of these different kinds of disturbances on the load side tracking performance with the basic controller structure (i.e., C, F_1, and F_2 in Fig. 1) are shown in Fig. 5. The four sharp peak errors corresponding to the four peak accelerations/decelerations in Fig. 4 are due to the model uncertainty in the model-based feedforward controller F_2. The oscillatory error at about 0.5–2 s and 2.5–4 s is resulted from the transmission error [15] at high (constant) speeds. The bump shape error at about 1.5–2.5 s is due to the interaction torque $d_{f\ell}$ from the load side disturbance setup, while the artificial

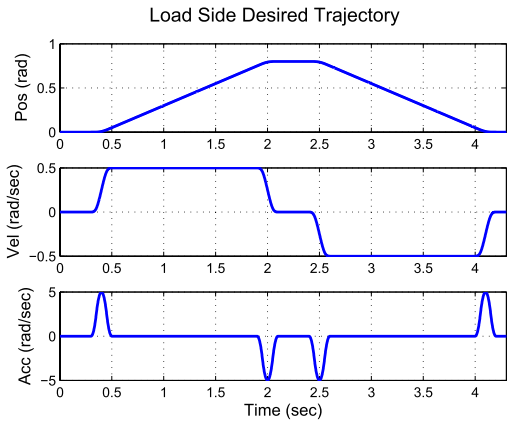


Fig. 4. Load side reference trajectory.

Fig. 5. Disturbance effects on load side position tracking error.

motor side disturbance d_{fm} results in the sinusoidal shape error from 3 s. The Coulomb friction causes the offset in the tracking error.

C. Algorithm Setup

The zero-phase acausal low-pass filters Q_r and Q_u are obtained as $Q_r(z) = Q_u(z) = Q_1(z^{-1})Q_1(z)$, where $Q_1(z)$ is a low-pass filter with a cutoff frequency of 30 Hz, which is beyond the system elastic antiresonant and resonant frequencies. With this selection of $Q_r(z)$ and $Q_u(z)$, the frequency responses of β_r in (16) and β_u in (22) with $\pm 50\%$ parametric uncertainties are plotted in Fig. 6 to verify the monotonic stability condition.

Fig. 6 shows that, using either motor side model or load side model, the magnitudes of β_r and β_u are always below 0 dB showing $\beta_r < 1$ and $\beta_u < 1$. Therefore, the monotonic stabilities (16) and (22) are ensured separately for both ILC schemes.

The two iteration varying gains in the hybrid dual-stage ILC scheme are tuned as

$$\gamma_{u,k} = \min\left(4\frac{\|e_{p,k}\|_2}{\|e_{p,1}\|_2}, 1\right) \tag{33a}$$

$$\gamma_{r,k} = 1 - \frac{1}{2}\gamma_{u,k} \tag{33b}$$

which follows the basic idea described in Section III-D.

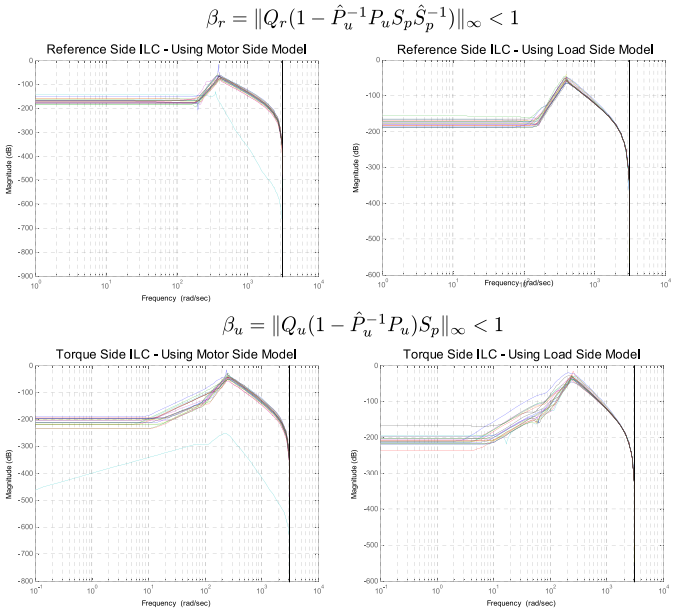


Fig. 6. Frequency responses of β_r and β_u with $\pm 50\%$ parametric uncertainties.

Now consider the dual-stage approach proposed in [7]. It can be shown that, the approach in [7] with plant inversion learning filters can be reformulated similarly as the reference ILC (L) with $\hat{P}_u = \hat{P}_{\ell u}$ plus the torque ILC (M) with $\hat{P}_u = \hat{P}_{\text{mu}}$ in this paper. This means that the dual-stage ILC scheme is performed with mismatched nominal models. As expected, this will not help to attenuate the model uncertainty but instead may even deteriorate the ILC convergence performance. To see this, the tracking performances in the following experiments are compared in three controller settings.

1) *RefILC(L):* Reference ILC only using load side learning, i.e., $\hat{P}_u(s) = \hat{P}_{\ell u}(s)$, $\gamma_{r,k} \equiv 1$, and $\gamma_{u,k} \equiv 0$.
2) *RefILC(L)+TrqILC(M):* Reference ILC using load side learning plus torque ILC using motor side learning, i.e., $\hat{P}_u(s) = \hat{P}_{\ell u}(s)$ for reference ILC and $\hat{P}_u(s) = \hat{P}_{\text{mu}}(s)$ for torque ILC. $\gamma_{r,k}$ and $\gamma_{u,k}$ are updated as in (33).
3) *RefILC(L)+TrqILC(L):* Reference ILC plus torque ILC both using load side learning, i.e., $\hat{P}_u(s) = \hat{P}_{\ell u}(s)$ for both reference ILC and torque ILC. $\gamma_{r,k}$ and $\gamma_{u,k}$ are updated as in (33)

where the continuous-time transfer functions are discretized by zero-order-hold method for the discrete-time signal processing.

D. Experimental Results

Each controller setting is implemented to track the load side reference trajectory in Fig. 4 for 10 iterations. First, the nominal model with accurately identified system dynamic parameters is used in the controller design. With an accurate nominal model, it is expected that the three controller settings will perform equally well since $\beta_r^* \approx 0$ in (16). As shown in Fig. 7, the load side position tracking errors for these three settings are all significantly reduced to almost the level of load side encoder resolution.

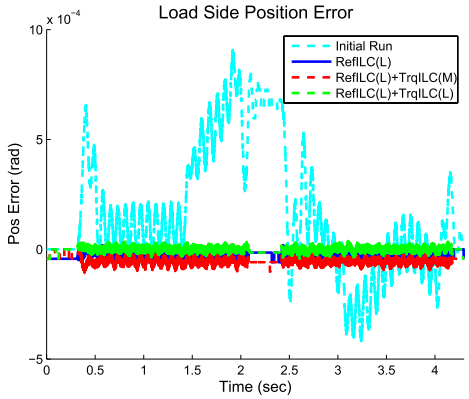
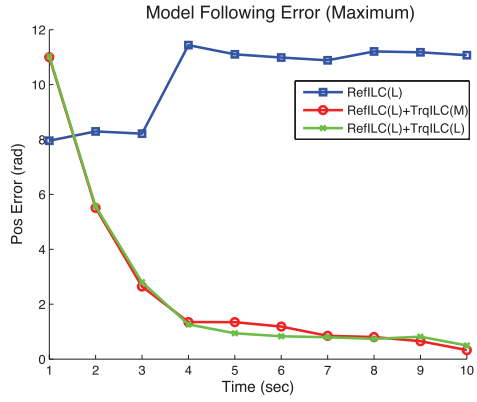
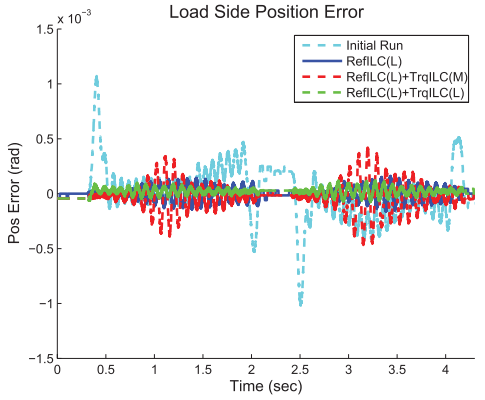


Fig. 7. Performance comparisons using accurate nominal model (after 10 iterations).



(a)



(b)

Fig. 8. Performance comparisons using nominal model with 15% parametric uncertainties (after 10 iterations). (a) Convergence of maximum model following error. (b) Load side position tracking error.

Next the three controller settings are compared using the nominal model with 15% uncertainty in the dynamic parameters. Normally, with larger model uncertainties, the cut off frequencies of Q filters need to be reduced to ensure the convergence of the learning process. Here, the Q filters are kept the same as the previous case to verify the benefits of the proposed scheme. Fig. 8 shows that the torque ILC performs very well once it is activated. No matter which nominal model is chosen to follow, the model following errors is greatly reduced with fast convergence rate. For the load side tracking performance, however, more differences are expected. It is shown in Fig. 8 that, due to the model uncertainty, the setting *RefILC(L)* does not perform as well as before. The setting *RefILC(L) + TrqILC(M)* actually deteriorates the performance because *TrqILC(M)* is intended to make the inner plant match with the motor side nominal model while the load side behavior may actually deviate further from its nominal behavior. In contrast, the setting *RefILC(L) + TrqILC(L)* performs the best since it intends to make the inner plant behave as the load side nominal model and thus greatly releases the uncertainty burden on the reference ILC.

V. Application to Multi-Joint Robots With Joint Elasticity

A more common example of the mismatched systems in industrial applications is an n-joint robot manipulator with gear compliance. Such robot can be considered as a system of n indirect drive trains connected in series resulting in a more complex mismatched dynamics. Normally, the robot is equipped with motor side encoders to provide direct measurements of motor side joint positions and velocities for real-time feedback, while the direct load side sensing at the joint is not available due to cost and assembly issues. However, it is relatively easy and beneficial [17] to have a three-axial accelerometer mounted at the robot end-effector to measure the end-effector acceleration in Cartesian space. This measurement can be further utilized offline to estimate the load side state information.

A. Robot Dynamic Model

1) Lagrangian Dynamics: The dynamics of the n-joint robot with joint compliance can be formulated as [3], [18]

$$M_\ell(q_\ell)\ddot{q}_\ell + C(q_\ell,\dot{q}_\ell)\dot{q}_\ell + G(q_\ell) + D_\ell\dot{q}_\ell + F_{\ell c}\mathrm{sgn}(\dot{q}_\ell) + J^{\mathrm{T}}(q_\ell)f_{\mathrm{ext}} = K_J\left(N^{-1}q_m - q_\ell\right) + D_J\left(N^{-1}\dot{q}_m - \dot{q}_\ell\right) \quad (34)$$

$$M_m\ddot{q}_m + D_m\dot{q}_m + F_{\mathrm{mc}}\mathrm{sgn}(\dot{q}_m) = \tau_m - N^{-1}\left[K_J\left(N^{-1}q_m - q_\ell\right) + D_J\left(N^{-1}\dot{q}_m - \dot{q}_\ell\right)\right] \quad (35)$$

where $q_\ell, q_m \in \mathbb{R}^n$ are the load side and the motor side position vectors, respectively, $\tau_m \in \mathbb{R}^n$ is the motor torque vector, $M_\ell(q_\ell) \in \mathbb{R}^{n\times n}$ is the load side inertia matrix, $C(q_\ell, \dot{q}_\ell) \in \mathbb{R}^{n\times n}$ is the Coriolis and centrifugal force matrix, and $G(q_\ell) \in \mathbb{R}^n$ is the gravity vector. M_m, K_J, D_J, D_ℓ, D_m, $F_{\ell c}$, F_{mc}, and $N \in \mathbb{R}^{n\times n}$ are all diagonal matrices. The (i,i)-th elements of these matrices, M_{mi}, K_{Ji}, D_{Ji}, $D_{\ell i}$, D_{mi}, $F_{\ell ci}$, F_{mci}, and N_i, represent the motor side inertia, joint stiffness, joint damping, load side damping, motor side damping, load side Coulomb friction, motor side Coulomb friction, and gear ratio of the i-th joint, respectively. $f_{\mathrm{ext}} \in \mathbb{R}^6$ denotes the external force acting on the robot due to contact with the environment. The matrix $J(q_\ell) \in \mathbb{R}^{6\times n}$ is the Jacobian matrix mapping from the load side joint space to the end-effector Cartesian space.

2) Decoupling Model: Define the nominal load side inertia matrix as $M_n = \mathrm{diag}(M_{n1}, M_{n2}, \ldots, M_{\mathrm{nn}}) \in \mathbb{R}^{n\times n}$, where $M_{ni} = M_{\ell,ii}(q_{\ell 0})$, and $M_{\ell,ii}(q_{\ell 0})$ is the (i,i)-th element of the inertia matrix $M_\ell(q_{\ell 0})$ at the home (or nominal) position

$q_{\ell 0}$. M_n can be used to approximate the inertia matrix $M_\ell(q_\ell)$. The off-diagonal entries of $M_\ell(q_\ell)$ represent the coupling inertia between the joints. Then, the robot dynamic model can be reformulated as follows:

$$M_m\ddot{q}_m + D_m\dot{q}_m = u + d^m(q) - N^{-1}\big[K_J\big(N^{-1}q_m - q_\ell\big) + D_J\big(N^{-1}\dot{q}_m - \dot{q}_\ell\big)\big] \quad (36a)$$

$$M_n\ddot{q}_\ell + D_\ell\dot{q}_\ell = d^\ell(q) + K_J\big(N^{-1}q_m - q_\ell\big) + D_J\big(N^{-1}\dot{q}_m - \dot{q}_\ell\big) \quad (36b)$$

where $u = \tau_m$ and $d^m(q) = -F_{\mathrm{mc}}\mathrm{sgn}(\dot{q}_m)$. All other coupling and nonlinear terms, such as Coriolis force, gravity, Coulomb frictions, and external forces, are grouped into a fictitious disturbance torque $d^\ell(q) \in \mathbb{R}^n$ as

$$\begin{aligned} d^\ell(q) = {} & \big[M_n M_\ell^{-1}(q_\ell) - I_n\big] \\ & \times\big[K_J\big(N^{-1}q_m - q_\ell\big) + D_J\big(N^{-1}\dot{q}_m - \dot{q}_\ell\big) - D_\ell\dot{q}_\ell\big] \\ & - M_n M_\ell^{-1}(q_\ell) \\ & \times\big[C(q_\ell,\dot{q}_\ell)\dot{q}_\ell + G(q_\ell) + F_{\ell c}\mathrm{sgn}(\dot{q}_\ell) + J^{\mathrm{T}}(q_\ell)f_{\mathrm{ext}}\big] \end{aligned} \quad (37)$$

where $q = [q_m^{\mathrm{T}},\ q_\ell^{\mathrm{T}}]^{\mathrm{T}}$ and I_n is an $n \times n$ identity matrix.

Thus, the robot can be considered as a MIMO system with $2n$ inputs and $2n$ outputs as in (3) and (4) with the fictitious disturbance input $d(j)$ defined as

$$d(j) = d(q(j)) = \big[[d^m(q(j))]^{\mathrm{T}},\ [d^\ell(q(j))]^{\mathrm{T}}\big]^{\mathrm{T}}. \quad (38)$$

Then similar to the single-joint case, the continuous time transfer functions from the inputs to the outputs for each joint can be derived from (36).

B. Robot Controller Structure

It is seen that the robot dynamic model (36) is in a decoupled form since all the variables are expressed in the diagonal matrix form or vector form. Therefore, the robot controller can be implemented in a decentralized form for each individual joint. Note that the MIMO linear system representation (3), (4) is obtained not through local linearization but by considering the fictitious disturbance d as an input that includes the model uncertainties and nonlinearities as well as coupling dynamics. The second component of the fictitious disturbance d in (38) influences the output in a different way from the motor torque input u. Thus this robot system is regarded as a MIMO mismatched dynamic system.

The controller structure to compensate for this fictitious disturbance d follows the one shown in Fig. 1 and (5) and (6). The initialization of the two feedforward updates for the first experiment iteration (i.e., initial run) is designed using nominal model as follows:

$$r_{q,1} = N\hat{K}_J^{-1}(\hat{\tau}_{\ell r} - \hat{M}_n\ddot{q}_{\ell r} - \hat{D}_\ell\dot{q}_{\ell r}) \quad (39)$$

$$\tau_{\mathrm{nl},1} = \tau_{\mathrm{ff},1} - \tau_{\mathrm{ln},1} \quad (40)$$

where the overall nominal feedforward torque $\tau_{\mathrm{ff},1}$ is computed by

$$\tau_{\mathrm{ff},1} = \hat{\tau}_{\mathrm{mr},1} + N^{-1}\hat{\tau}_{\ell r} \quad (41)$$

$$\begin{aligned} \hat{\tau}_{\ell r} = {} & \hat{M}_\ell(q_{\ell r})\ddot{q}_{\ell r} + \hat{C}(q_{\ell r},\dot{q}_{\ell r})\dot{q}_{\ell r} + \hat{G}(q_{\ell r}) + \hat{D}_\ell\dot{q}_{\ell r} \\ & + \hat{F}_{\ell c}\mathrm{sgn}(\dot{q}_{\ell r}) + J^{\mathrm{T}}(q_{\ell r})f_{\mathrm{ext},r} \end{aligned} \quad (42)$$

$$\hat{\tau}_{\mathrm{mr},1} = \hat{M}_m\ddot{\bar{q}}_{\mathrm{mr},1} + \hat{D}_m\dot{\bar{q}}_{\mathrm{mr},1} + \hat{F}_{\mathrm{mc}}\mathrm{sgn}(\dot{\bar{q}}_{\mathrm{mr},1}). \quad (43)$$

Remark 10: Note that (41)–(43) are obtained from (34) and (35) using the nominal model representations. With this nominal torque calculation, $r_{q,1}$ in (39) aims to account for the joint torsion resulted from the nominal nonlinear reference torque at the load side.

From (40) and Fig. 1, it is seen that the overall nominal feedforward torque $\tau_{\mathrm{ff},1}$ is divided into two parts. The nonlinear part $\tau_{\mathrm{nl},1}$ is injected inside the inner plant to reduce model uncertainties and to make the inner plant behave as the chosen nominal linear model. With this in mind, the linear feedforward torque $\tau_{\mathrm{ln},1}$, computed by (6) using the nominal linear model, can be injected outside the inner plant to achieve the nominal performance.

C. Hybrid Scheme for Dual-Stage ILC

With the above model formulation, the dual-stage ILC scheme (29) and (30) can be implemented. However, because of the sensor configuration, the load side joint position can only be estimated rather than the actual encoder clean measurement as in the single-joint setup. In this non-ideal sensing case, the plant inversion in (21) usually encounters numerical difficulty since the relative orders of $P_{\ell u}(s)$ and $P_{\mathrm{mu}}(s)$ are 3 and 2, respectively. Thus it is more favorable to choose $P_u(s) = P_{\ell u}(s)s^2$ or $P_u(s) = P_{\mathrm{mu}}(s)s$, both of which have lower relative orders. The corresponding desired learning information for this new choice of P_u is the load side acceleration $\ddot{q}_{\ell,k}$ or the motor side velocity $\dot{q}_{m,k}$, which is available with the assumed sensor configuration (i.e., end-effector accelerometer and motor encoders). By similar derivation, the torque ILC scheme with these changes can still be obtained exactly the same as in Section III-C, and achieves the same objective, i.e., making the inner plant behave as the chosen nominal model. Another benefit of this modification is that the torque ILC will be effective for vibration suppression, while the reference ILC achieves the position error reduction. Thus, in the practical implementation such as the following experimental study, these changes will be applied to the torque ILC scheme to avoid numerical instability.

Similar to the single-joint case, the disturbance d is state dependent, and is repetitive only if the robot state q is repetitive. Since the robot basic performance should be already close to satisfactory, the tiny changes of q around the reference state q_r in each iteration normally will not result in drastic change in d_k (i.e., the repetitiveness of d_k can be approximated). However, the interference between the two ILC stages still requires the hybrid scheme (29), (30) to be implemented. More specifically, the two gains $\gamma_{u,k}^i$ and $\gamma_{r,k}^i$ for the i-th joint are tuned by trial and error

$$\gamma_{u,k}^i = \max\left(0.2, \min\left(\frac{6}{\gamma_{u,k-1}^i}\left|\frac{\|e_{p,k}^i\|_2}{\|e_{p,k-1}^i\|_2} - 1\right|, 1\right)\right) \quad (44a)$$

$$\gamma_{r,k}^i = \left(1 - 0.8\gamma_{u,k}^i\right)\cdot\min\left(2\cdot\frac{\|e_k^i\|_\infty}{\|e_1^i\|_\infty}, 1\right) \quad (44b)$$

with the initialization as $\gamma_{u,1}^i = 1$, $\gamma_{r,1}^i = 0.2$. In addition to the tradeoff between the torque ILC and the reference ILC as discussed in Section III-D, two other considerations are taken

here. First, if the maximum tracking error is sufficiently small (i.e., $\|e_k^i\|_\infty/\|e_1^i\|_\infty \approx 0$), the reference ILC becomes less necessary. Thus, the gain $\gamma_{r,k}^i$ is accordingly decreased. On the other hand, to maintain the basic convergence rate, the gain $\gamma_{u,k}^i$ for the torque ILC is constrained to be within [0.2, 1] as shown in (44). Note that these two gains may not be optimal and can be improved at a cost of complexity if we further consider the coupling effects between the joints. However, this consideration may not be necessary. In the experiments, we have shown that (44) is able to perform well for a number of trajectories with this relatively simple form.

D. Robot Load Side State Estimation

As mentioned above, the required load side joint information (i.e., $q_{\ell,k}$ and $\ddot{q}_{\ell,k}$) for load side learning cannot be measured directly. Therefore, it is desired to retrieve this information from the available sensing, i.e., by fusing the measured end-effector acceleration with the motor encoder measurements. This estimation problem is addressed here by utilizing the scheme developed in [17].

With the robot dynamic model (35), the load side joint position q_ℓ can be roughly estimated as

$$\hat{q}_\ell^o = (\hat{D}_J s + \hat{K}_J)^{-1}\Big[\hat{K}_J N^{-1} q_m + \hat{D}_J N^{-1}\dot{q}_m - N \times \Big(\tau_m - \hat{M}_m \hat{\ddot{q}}_m - \hat{D}_m \dot{q}_m - \hat{F}_{\rm mc}\,\mathrm{sgn}(\dot{q}_m)\Big)\Big] \tag{45}$$

where q_m and $\dot{q}_m$ are obtained from motor encoder measurements, and τ_m can be either motor torque command or measured by motor current. The reference trajectory $\ddot{q}_{\rm mr}$ is used instead of $\hat{\ddot{q}}_m$ in (45) as approximation. Furthermore, with Euler differentiation of $\hat{q}_\ell^o$, the rough estimate of the load side joint velocity, $\hat{\dot{q}}_\ell^o$, is obtained.

The optimal load side joint acceleration estimate is then obtained by solving the following least squares problem:

$$\min_{\hat{\ddot{q}}_\ell(j)} \quad f(\hat{\ddot{q}}_\ell(j)) = \frac{1}{2}\left\|\sum_{i=0}^{j}\hat{\ddot{q}}_\ell(i)\Delta t - \hat{\dot{q}}_\ell^o(j)\right\|_2^2$$
$$\text{s.t.} \quad \bar{J}(q_\ell(j))\hat{\ddot{q}}_\ell(j) = \ddot{p}_e(j) - \dot{\bar{J}}(q_\ell(j), \dot{q}_\ell(j))\dot{q}_\ell(j) \tag{46}$$

where $\bar{J}(q_\ell)$ and $\dot{\bar{J}}(q_\ell)$ are the first three rows of the corresponding Jacobian matrices $J(q_\ell)$ and $\dot{J}(q_\ell)$, respectively, Δt is the sampling time, and $\ddot{p}_e$ is the computed acceleration measurements in the world coordinates with gravity removed. The closed-form solution of this optimization problem is documented in [17]. Then the refinement of the joint space position estimate $\hat{q}_\ell$ can be obtained through the decoupled kinematic Kalman filter for each joint, using expectation-maximization (EM) for the covariance adaptation (see [17] for details).

E. Experimental Study

1) Test Setup: The proposed method is implemented on a six-joint industrial robot, FANUC M-16*i*B/20 (Fig. 9), for end-effector position tracking and vibration reduction. The robot is equipped with built-in motor encoders for each joint.

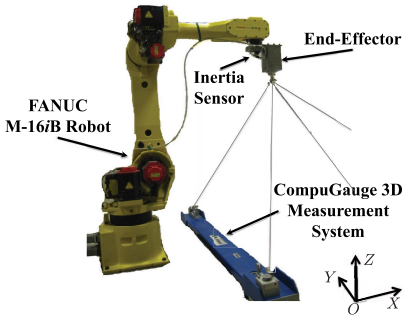


Fig. 9. FANUC M-16*i*B robot system.

An inertia sensor (Analog Devices, ADIS16400) containing a three-axial accelerometer is attached to the end-effector. The 3-D position measurement system, CompuGauge 3D (repeatability of 0.02 mm, accuracy of 0.15 mm, resolution of 0.01 mm), is used to measure the end-effector tool center point (TCP) position as a ground truth for performance validation. The sampling rates of all the sensor signals and the real-time controller implemented through MATLAB xPC Target are set to 1 kHz. System identification experiments are conducted for each individual joint at several different postures to obtain the nominal dynamic parameters in the dynamic model (36).

2) Algorithm Setup: The zero-phase acausal low-pass filters Q_r and Q_u are obtained as $Q_r(z) = Q_u(z) = Q_1(z^{-1})Q_1(z)$, where $Q_1(z)$ is a diagonal matrix of low-pass filters with cutoff frequencies beyond the identified first elastic antiresonant frequency of the corresponding joint to deal with the joint elasticity. With this selection of $Q_r(z)$ and $Q_u(z)$, the frequency responses of β_r in (16) and β_u in (22) using load side inertia variations among the workspace are checked to verify the monotonic stability conditions.

To see the superiority of proposed methods (i.e., hybrid dual-stage scheme versus single stage scheme, load side learning versus motor side learning), the tracking performances in the experiments are compared in the four controller settings implemented for 10 iterations each as follows.

1) *RefILC(L):* Reference ILC only using load side learning, i.e., $\hat{P}_u(s) = \hat{P}_{\ell u}(s)$, $\gamma_{r,k} \equiv 1$, and $\gamma_{u,k} \equiv 0$.
2) *TrqILC(L):* Torque ILC only using load side learning, i.e., $\hat{P}_u(s) = \hat{P}_{\ell u}(s)s^2$, $\gamma_{r,k} \equiv 0$, and $\gamma_{u,k} \equiv 1$.
3) *RefILC(L)+TrqILC(L):* Reference ILC plus torque ILC using load side learning, i.e., $\hat{P}_u(s) = \hat{P}_{\ell u}(s)$ for reference ILC and $\hat{P}_u(s) = \hat{P}_{\ell u}(s)s^2$ for torque ILC. $\gamma_{r,k}$ and $\gamma_{u,k}$ are updated as in (44).
4) *RefILC(M)+TrqILC(M):* Reference ILC plus torque ILC using motor side learning, i.e., $\hat{P}_u(s) = \hat{P}_{\rm mu}(s)$ for reference ILC and $\hat{P}_u(s) = \hat{P}_{\rm mu}(s)s$ for torque ILC. $\gamma_{r,k}$ and $\gamma_{u,k}$ are updated as in (44)

where the continuous-time transfer functions are discretized by zero-order-hold method for the discrete-time signal processing.

3) Experimental Results: The testing TCP trajectory (Fig. 10) is a 10 cm × 10 cm square path on the Y–Z plane with fixed orientation, maximum velocity of 1 m/s, and maximum acceleration of 12.5 m/s^2. In addition to the reference trajectory (*Reference*), Fig. 10 also shows the actual

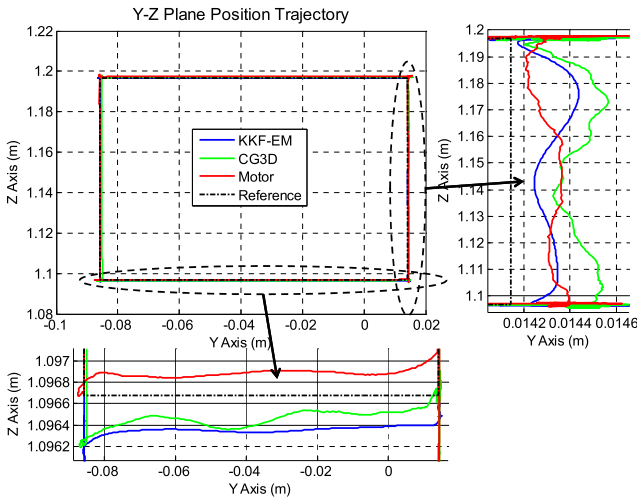


Fig. 10. $Y-Z$ plane TCP position estimation (experiment).

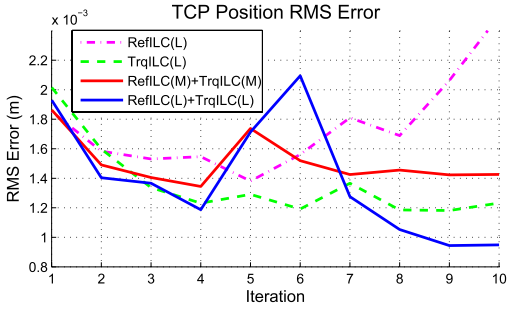


Fig. 11. TCP position RMS error comparisons in iteration domain.

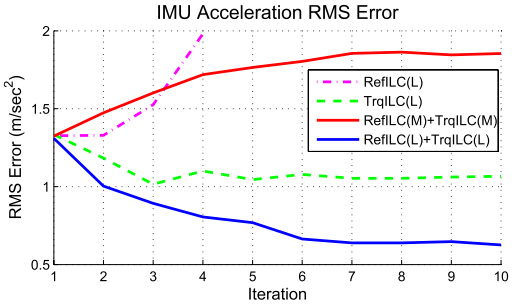


Fig. 12. IMU acceleration RMS error comparisons in iteration domain.

trajectory (*CG3D*) measured by CompuGauge and the estimated trajectory by proposed method (*KKF-EM*) or by motor encoder measurements only (*Motor*). It is seen that the *KKF-EM* estimation performs much better than the *Motor* setting by capturing closer transient motion on the Y axis and with much less offset on the Z-axis.

Figs. 11 and 12 show the iteration domain root-mean-square (RMS) tracking error[5] convergence profiles, while Figs. 13 and 14 show the time domain error profiles of the initial run and the last runs of these controller settings.

[5]The Cartesian space error here is defined as the Euclidean distance between the estimated position/acceleration and the actual measured position/acceleration.

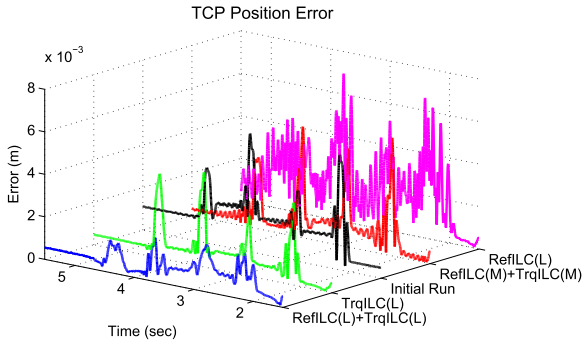


Fig. 13. TCP position error comparisons in time domain (initial run versus the 10th iteration of four controller settings).

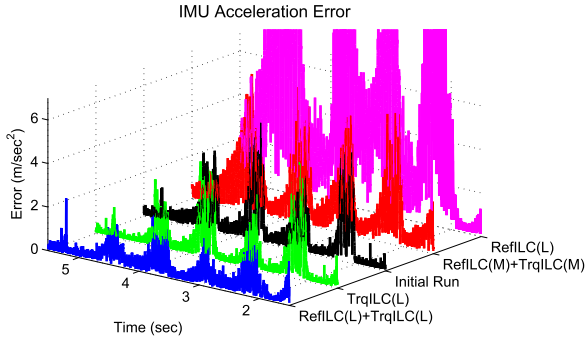


Fig. 14. IMU acceleration error comparisons in time domain (initial run versus the 10th iteration of four controller settings).

It can be seen that the *RefILC(L)* + *TrqILC(L)* setting achieves the overall best performance in position tracking (see TCP position error) and vibration reduction (see IMU acceleration error), even though there is non-monotonic transient around the fifth and the sixth iterations in the position error convergence due to the interference between the two ILC stages. The *RefILC(L)* setting turns out to be unstable in the iteration domain without the help of torque ILC to reduce mismatched model uncertainty. This implies that the Q filter bandwidth (i.e., learning ability) for the *RefILC(L)* setting needs to be further compromised. The *TrqILC(L)* setting looks monotonically convergent but with quite limited improvement in error reductions (especially for moving periods), since the torque ILC aims at model matching for the inner plant and does not directly address the load side tracking error. The *RefILC(M)* + *TrqILC(M)* setting does not perform well either, since motor side model can only be used for motor side learning, while the load side (end-effector) performance is not guaranteed and may be even degraded due to the mismatched dynamics.

VI. Conclusion

In this paper, a model-based dual-stage ILC scheme was developed for a class of MIMO mismatched systems. To improve the performance of the ILC stage aiming for tracking error reduction, another ILC utilizing the idea of model following was designed to drive the inner plant to

behave like the nominal model. The convergence property and performance bandwidth were emphasized. To make the two ILC stages work together, an *ad hoc* hybrid scheme was proposed to make transitions between the two ILC stages. A single-joint indirect drive system with several inherent and designed external disturbances was utilized to experimentally validate the proposed ILC scheme. The proposed scheme was also applied to multi-joint robots with joint elasticity. The experimental validation on a six-joint industrial robot has demonstrated the advantage of the hybrid dual-stage load learning scheme to deal with the mismatched dynamics for end-effector position tracking and vibration mitigation.

REFERENCES

[1] D. A. Bristow and M. Tharayil, "A survey of iterative learning control: A learning-based method for high-performance tracking control," *IEEE Control Syst. Mag.*, vol. 26, no. 3, pp. 96–114, Jun. 2006.

[2] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *J. Robot. Syst.*, vol. 1, no. 2, pp. 123–140, 1984.

[3] M. W. Spong, "Modeling and control of elastic joint robots," *J. Dyn. Syst., Meas., Control*, vol. 109, no. 4, pp. 310–318, Dec. 1987.

[4] H. K. Khalil, *Nonlinear Systems*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2002.

[5] J. K. Hedrick and P. P. Yip, "Multiple sliding surface control: Theory and application," *J. Dyn. Syst., Meas., Control*, vol. 122, no. 4, pp. 586–593, Feb. 2000.

[6] B. Yao and M. Tomizuka, "Adaptive robust control of MIMO nonlinear systems in semi-strict feedback forms," *Automatica*, vol. 37, no. 9, pp. 1305–1321, Sep. 2001.

[7] F. Miyazaki, S. Kawamura, M. Matsumori, and S. Arimoto, "Learning control scheme for a class of robot systems with elasticity," in *Proc. 25th IEEE Conf. Decision Control*, vol. 25. Dec. 1986, pp. 74–79.

[8] M. Wada, T. Tsukahara, K. Tsuda, F. Electric, and E. D. Division, "Learning control of elastic joint robot and its application to the industrial robot manipulator," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1993, pp. 417–422.

[9] D. De Roover and O. Bosgra, "Synthesis of robust multivariable iterative learning controllers with application to a wafer stage motion system," *Int. J. Control*, vol. 73, no. 10, pp. 968–979, 2000.

[10] D. Owens, J. Hatonen, and S. Daley, "Robust monotone gradient based discrete time iterative learning control," *Int. J. Robust Nonlinear Control*, vol. 19, no. 6, pp. 634–661, Apr. 2009.

[11] W. Hakvoort, R. Aarts, J. van Dijk, and J. Jonker, "Model-based iterative learning control applied to an industrial robot with elasticity," in *Proc. IEEE CDC*, Dec. 2007, pp. 4185–4190.

[12] W. Chen and M. Tomizuka, "A two-stage model based iterative learning control scheme for a class of MIMO mismatched linear systems," in *Proc. ISFA*, Jun. 2012, pp. 159–166.

[13] W. Chen and M. Tomizuka, "Iterative learning control with sensor fusion for robots with mismatched dynamics and mismatched sensing," in *Proc. DSCC*, Oct. 2012, pp. 1480–1488.

[14] F. Padieu and R. Su, "An H_∞ approach to learning control systems," *Int. J. Adapt. Control Signal Process.*, vol. 4, no. 6, pp. 465–474, Nov. 1990.

[15] C.-H. Han, C.-C. Wang, and M. Tomizuka, "Suppression of vibration due to transmission error of harmonic drives using peak filter with acceleration feedback," in *Proc. 10th IEEE Int. Workshop Adv. Motion Control*, Mar. 2008, pp. 182–187.

[16] P. Lambrechts, M. Boerlage, and M. Steinbuch, "Trajectory planning and feedforward design for electromechanical motion systems," *Control Eng. Pract.*, vol. 13, no. 2, pp. 145–157, Feb. 2005.

[17] W. Chen and M. Tomizuka, "Direct joint space state estimation in robots with multiple elastic joints," *IEEE/ASME Trans. Mechatron.* [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6502721

[18] C. C. de Wit, G. Bastin, and B. Siciliano, *Theory of Robot Control*, 1st ed. New York, NY, USA: Springer-Verlag, Jan. 1996.

Wenjie Chen (S'12–M'12) received the B.Eng. degree from Zhejiang University, Zhejiang, China, in 2007, and the M.S. and Ph.D. degrees from the University of California, Berkeley, CA, USA, in 2009 and 2012, respectively, all in Mechanical Engineering.

He is currently a Post-Doctoral Scholar with the Department of Mechanical Engineering, the University of California, Berkeley. His current research interests include design and implementation of advanced control and sensing algorithms with applications to robotic/mechatronic systems, such as industrial robots, wearable assistive robotics, and robots for advanced manufacturing.

Masayoshi Tomizuka (M'86–F'97) received the Ph.D. degree in Mechanical Engineering from the Massachusetts Institute of Technology, in 1974.

He joined the Department of Mechanical Engineering, University of California, Berkeley, in 1974, where he is currently the Cheryl and John Neerhout, Jr., Distinguished Professor. He conducts research on optimal and adaptive control, digital control, motion control, and their applications to robotics, manufacturing, information storage devices, and vehicles. He served as the Program Director of the Dynamic Systems and Control Program of the National Science Foundation from 2002 to 2004. He was a Technical Editor of the *ASME Journal of Dynamic Systems, Measurement and Control* and the Editor-in-Chief of the IEEE/ASME TRANSACTIONS ON MECHATRONICS. He was the recipient of the Rudolf Kalman Best Paper Award (ASME, 1995, 2010), the Charles Russ Richards Memorial Award (ASME, 1997), the Rufus Oldenburger Medal (ASME, 2002), and the John R. Ragazzini Award (American Automatic Control Council, 2006).