2017 11th Asian Control Conference (ASCC)
Gold Coast Convention Centre, Australia
December 17-20, 2017

Design of Feedback Gain in Feedback-Based Iterative Learning Control

Gijo Sebastian, Ying Tan, Denny Oetomo and Iven Mareels

***Abstract*— Feedback loops have been introduced in iterative learning control (ILC) to improve the performance in the time domain. This in turn affects the convergence performance in the iteration domain. Such an impact on ILC performance has not been rigorously analysed in the literature. In this paper, the concept of dynamic influence in the iteration domain is introduced to characterise the impact of feedback in the convergence of ILC algorithms. An optimal feedback gain is thus designed in order to minimise such an impact with the help of nominal model, leading to an improvement of the transient response in the iteration domain. The result also reveals a clear performance trade-off between the robustness of the system in time domain and the transient response in iteration domain.**

I. INTRODUCTION

Iterative learning control (ILC) is a technique used for systems with repetitive task executions where the control input is updated in every trial so as to improve the tracking performance. It finds application in batch manufacturing, chemical processing, precision motion control, human motor learning models and robotic rehabilitation [1]–[5]. It is well-known that the design of ILC has a clear hybrid nature: a finite time domain (continuous-time/discrete-time) and an infinite iteration domain. This nature makes ILC analysis distinct from other control design methodologies in which stability in time domain is the major consideration. The various developments of ILC algorithms are summarized in the survey papers [6]–[8]. In [9], the history of ILC starting from [10] and the basic analysis tools have been summarized.

Many ILC algorithms depend only on the error from previous cycles and the direct implementation of such ILC algorithms has a feed-forward nature as the control input is pre-calculated before executing a trial. There are many ILC algorithms that use the current error signal together with the error from previous trials, leading to so called feedback-based ILC design as shown in [11]. Normally in the design of a feedback-based ILC framework, specific roles are assigned to two controllers without investigating their interactions [12]–[16]. Usually, a feedback controller is expected to deal with internal state domain, non-repeating disturbances, and maintain robustness whereas the ILC improves the performance of tracking in iterations even if the feedback controller is poorly designed. As both ILC and feedback control improves tracking performance using the error information, it is evident that the learning behaviour, in particular, the transient performance in the iteration domain will be influenced by the selection of the feedback controller.

Melbourne School of Engineering, The University of Melbourne, Parkville VIC 3010 Australia {gijos, yingt, doetomo,i.mareels}@unimelb.edu.au

In most of the literature in the analysis of feedback-based ILC schemes, the design of feedback loop and ILC loop are decoupled. None of these analyses have addressed the impact of feedback in the transient response in the iteration domain, though convergence is always guaranteed. The effect of the system dynamics, which is either open loop or closed loop with a feedback controller, is often neglected in the analysis of ILC. The feedback and ILC interactions were pointed out in [17] where it is observed that the bandwidth of feedback controller influences the transient behaviours in ILC. A design trade-off between convergence in the iteration domain and robustness in the time domain in frequency domain analysis was established in [18], [19]. Different from the literature, in this paper the feedback is utilised to improve the transient responses of ILC in iteration domain, instead of using it for robustness. To do so, the interactions between the design of the feedback controller and ILC will be investigated.

A new performance index called dynamic influence in the iteration domain is introduced to capture the transient behaviour in the iteration domain. It is noted as in Property 1 that the dynamic influence in the iteration domain is related to the stability margin in the time domain. It is interesting to show that a large stability margin might not lead to a minimal dynamic influence in iteration domain, which is consistent with the observations in literature as [18], [19].

Once the dynamic influence in the iteration domain is introduced, the role of feedback is thus to minimize this new cost. Either model-based off-line optimization or on-line model-free optimization techniques can be used to find an optimal feedback gain to minimize this dynamic influence. By using optimal feedback gain, the transient response in the iteration domain can be greatly improved.

It is worthwhile to highlight, though this paper only optimises the transient response in the iteration domain by selecting the feedback gain, the same idea can be extended to performances in both the time domain and iteration domain by tuning the feedback gain and the updating gain in ILC simultaneously. Hence this work presents a new design framework for incorporating two design freedoms at the same time to achieve a better performance.

The reminder of this paper is organized as follows. Section II provides preliminaries, problem formulation and a motivating example, followed by main results in Section III. Simulations are presented in Section IV while Section V concludes the paper.

978-1-5090-1573-3/17/$31.00 ©2017 IEEE

II. PRELIMINARIES, PROBLEM FORMULATION AND MOTIVATION

The following notations are used in this paper. The notation $\mathcal{R}$ denotes a set containing real numbers and $\mathcal{N}$ stands for the set containing the non-negative integers. The notation $\mathcal{C}[0,\ T]$ represents the set of all continuous functions defined on $[0,\ T]$, while $\mathcal{C}^1[0,\ T]$ represents the set of all continuous and differentiable functions with derivatives defined on $\mathcal{C}[0,\ T]$. Let $\mathbf{f}(t) \in \mathcal{C}[0,\ T]$, then $|\mathbf{f}(t)|$ denotes its Euclidean norm at the time instance t, and $\|\mathbf{f}\|_s$ denotes its supremum norm, i.e., $\|\mathbf{f}\|_s = \max\limits_{t\in[0,\ T]} |\mathbf{f}(t)|$. Let λ be a positive constant. The λ-norm of $\mathbf{f}$ is defined as $\|\mathbf{f}\|_\lambda = \max_{t\in[0,\ T]} e^{-\lambda t}|\mathbf{f}(t)|$.

For simplicity, for any matrix $A \in \mathcal{R}^{n\times n}$, we denote

$$\lambda_R(A) = \max_{i=1,\dots,n}\{Re(\lambda_i)\}, \tag{1}$$

where λ_i is the i^{th} eigenvalue of A and $Re(\cdot)$ denotes the real part of the complex number.

Let $\mu > 0$ be any positive constant. Let A_J be the Jordan form of a square matrix A. A modified Jordan form $A_{J,\mu}$ with respect to μ is defined as a matrix which is identical to A_J except that each off diagonal element "1" is replaced by $\frac{1}{\mu}$ or off diagonal I is replaced by $\frac{1}{\mu}I$ [20, P43,Theorem 2.2.7].

Lemma 1 is needed to estimate the upper bound of trajectories of ILC algorithms in time domain.

Lemma 1: Let $\mu > 0$, for any matrix $A \in \mathcal{R}^{n\times n}$ there exists a non-singular matrix $T_\mu \in \mathcal{R}^{n\times n}$ such that

$$T_\mu e^{At} T_\mu^{-1} = e^{A_{J,\mu}t}, \quad \forall t \geq 0. \tag{2}$$

Moreover,

$$\left|T_\mu e^{At} T_\mu^{-1}\right| = \left|e^{J_\mu t}\right| \leq e^{(\lambda_R(A)+\frac{1}{\mu})t}, \quad \forall t \geq 0. \tag{3}$$

where $\lambda_R(A) + \frac{1}{\mu} < 0$

Proof: The proof is given in [21, Lemma-1]

A. Plant Model

For simplicity of presentation, this paper considers a single-input-single-output (SISO) plant that can be represented in state space as:

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + Bu \\ y &= C\mathbf{x}\ ; \end{aligned} \tag{4}$$

where $\mathbf{x} \in \mathcal{R}^n$, $u \in \mathcal{R}$ and $y \in \mathcal{R}$ are the state, input and output vectors respectively. The state matrices (A, B, C) are of appropriate dimensions. Even though the system under consideration is SISO, the analysis presented in this paper can be easily extended to multiple -input -multiple-output (MIMO) square system[1].

Assumption 1: It is assumed that the system (A, B) is controllable and (A, C) is observable.

The condition (A, B) is controllable indicates that the feedback has enough design freedom such that the closed loop eigenvalues can be arbitrarily designed. (A, C) is observable indicates that the state can be estimated from appropriately designed observer. For simplicity of presentation, it is assumed that the state of the system (4) is measurable.

[1]A MIMO square system is defined as the system which has the same dimension for input and output vectors

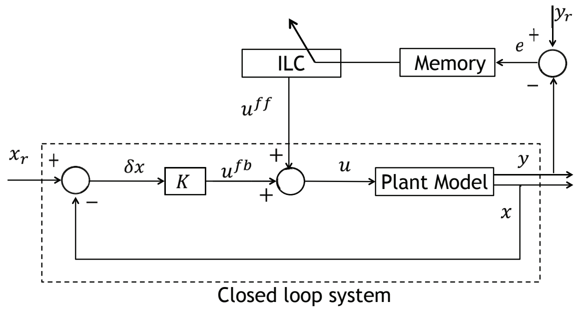

Fig. 1. A feedback-based ILC system

Assumption 2: For every reference trajectory $y_r \in \mathcal{C}^1[0,\ T]$, there exist $\mathbf{x}_r \in \mathcal{C}^1[0,\ T]$ and $u_r \in \mathcal{C}[0,\ T]$ that satisfy the following dynamics:

$$\begin{aligned} \dot{\mathbf{x}}_r &= A\mathbf{x}_r + Bu_r \\ y_r &= C\mathbf{x}_r\ . \end{aligned} \tag{5}$$

Assumption 2 is a matching condition to ensure that the desired trajectory is reachable for the given systems. This is a typical setting in model reference tracking.

The tracking error, $e(t)$ is defined as:

$$e(t) = y_r(t) - y(t),\ \forall t \in [0,\ T]. \tag{6}$$

Assumption 3: The system has a relative degree one such that $CB > 0$.

Assumption 3 can be relaxed. The same analysis can be applied to systems with a higher relative degree.

The control objective is to track the desired reference y_r perfectly when the tracking task is repeated and the iteration number tends to infinity, *i.e.*,

$$\lim_{i\to\infty} |e_i(t)| = 0 \quad \forall t \in [0, T], \tag{7}$$

where $e_i(t) = r(t) - y_i(t)$. Here the subscript $(\cdot)_i$ is used to indicate the iteration number. For example, $y_i(t)$ is the output of the plant (4) at the i^{th} iteration.

Assumption 4: For a given $y_r(t) \in \mathcal{C}^1[0, T]$, the system (4) satisfies the identical initial condition (*i.i.c*) represented by $\mathbf{x}_i(0) = \mathbf{x}_r(0), \quad \forall i \in \mathcal{N}$.

Assumption 4 is needed in order to achieve perfect tracking performance [22].

B. Controller Design

This subsection summarizes the typical design strategies for the feedback-based ILC algorithm. The block diagram of the feedback-based ILC design is shown in Fig.1. It consists of a dual controller design which is composed of a stabilising feedback controller (not tracking) and an ILC system based on the output tracking error.

The control law is defined as:

$$u_i(t) = u_i^{fb}(t) + u_i^{ff}(t), \quad \forall t \in [0, T], \tag{8}$$

where $u_i^{fb}(t)$ is a stabilising feedback controller given in (9) and u_i^{ff} is the ILC control input (also refereed as feed-forward input) given in (10)

Without the loss of generality, we assume a state feedback stabilisation of the form:

$$u_i^{fb} = K(\mathbf{x}_r - \mathbf{x}) \tag{9}$$

where the feedback gain, K is designed to stabilize the system (4). In other words, the stabilizing controller will ensure that the state stays close to the desired state x_r.We denote the set S as the set containing all stabilizing K matrices.

We use a standard D-type ILC represented by:

$$u_{i+1}^{ff}(t) = u_i^{ff}(t) + \gamma \dot{e}_i(t), \qquad u_1^{ff}(t) = 0, \quad \forall t \in [0,\ T]. \tag{10}$$

where γ represents the learning gain.

For such a dual controller design, the error dynamics for the closed loop system can be written in the form:

$$\begin{aligned} \delta\dot{\mathbf{x}} &= (A - BK)\,\delta\mathbf{x} + B\delta u^{ff} \\ \delta y &= C\delta\mathbf{x} \end{aligned} \tag{11}$$

where $\delta\mathbf{x} = \mathbf{x}_r - \mathbf{x}$ and $\delta u^{ff} = u_r - u^{ff}$.

For any reference signal $y_r(t)$, if Assumptions 2-4 hold and the following convergence condition is satisfied [22]

$$|1 - \gamma CB| \leq \rho < 1, \tag{12}$$

then $\lim_{i\to\infty} \|e_i\|_\lambda = \lim_{i\to\infty} \|e_i\|_s = 0$.

Remark 1: It is noted that the ILC updating law (10) can be applied to a large family of systems that satisfy the convergence condition. As it only requires the "sign" and bounds of CB in the design, it is a kind of model-free controller. If the upper and lower bounds of CB are known, the learning gain γ can be designed accordingly. If $CB = 0$, higher order derivatives of the tracking error are needed in the updating law [2] and a slightly different convergence condition is needed to ensure the convergence. ◦

Remark 2: In the proof of the convergence [22], the time-weighted norm or λ-norm is used with the help of Gronwall Lemma. The key idea in the proof is to ignore the dynamics of the system by using a λ-norm with a very large constant λ. No matter whether the matrix A is stable or not, when a finite-time interval is considered, the trajectories of the system (4) are always bounded provided that the control input is bounded. Thus by designing the updating law (10), a contraction mapping in terms of $\|\dot{e}_i\|_\lambda$ will be ensured provided that the convergence condition (12) is satisfied with the resetting condition given in Assumption 4. However, even though the convergence of a simple ILC (10) can be guaranteed, the transient performance along iteration domain might be very bad as noted in [22]. ◦

TABLE I

FEEDBACK GAINS AND POLE PLACEMENT LOCATIONS

	Case-1	Case-2	Case-3	Case-4
Closed loop Poles	$-2 \pm j$	$-3 \pm j$	$-4 \pm j$	$-5 \pm j$
K	[6 3.99]	[11 5.99]	[18 7.99]	[27 9.99]

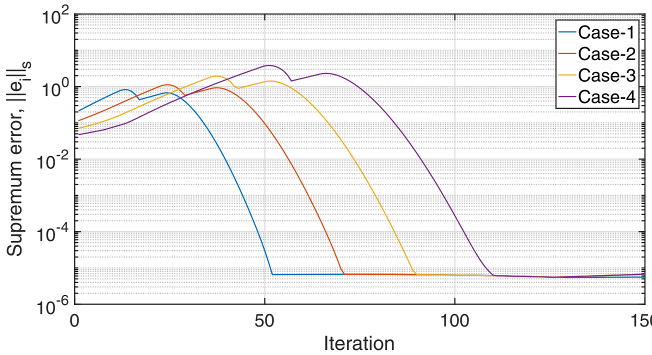

Fig. 2. Typical example of variation of supremum error in feedback-based ILC system where $u_i = u_i^{fb} + u_i^{ff}$

C. A Motivating Example

It seems intuitively clear that a stabilising feedback control will improve the transient response in the time domain as well as in the iteration domain [23]. However, this is not always the case as illustrated in the following example.

Consider a linear time invariant plant model:

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 1 & -0.01 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ y &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{aligned} \tag{13}$$

It can be verified that the system satisfies the Assumption-1 and the reference state, $\mathbf{x}_r(t) = [0.4\sin(\frac{2\pi t}{T}),\ \frac{0.8\pi}{T}\cos(\frac{2\pi t}{T})]$ is as per Assumption-2.

Four different cases of feedback gains for the controller (9) are used in the motivation example as tabulated in Table I. A time step of 5×10^{-4} is used for simulation with initial ILC input $u_1^{ff}(t) = 0$ and learning rate $\gamma = 0.9$ in which the convergence condition is satisfied. The time derivative of er-

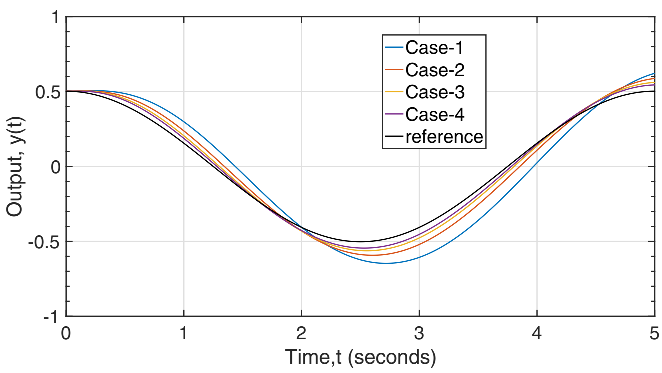

Fig. 3. Output response when $u_1^{ff} = 0$

ror is calculated by using a backward difference method and using a butter-worth filter of order-3 with cut off frequency of $300Hz$. The location of closed loop poles (as per Table I) are used to find the feedback gains that affect the performance in iteration domain, though the convergence condition is not affected by the feedback.

It can be seen in Fig.2 that all four cases have different iteration domain performance. Case-1 has the fastest convergence speed in terms of number of iterations to reach the lowest possible value of error in simulations where the initial error for Case-1 in the first iteration is more than that of other cases considered.This shows that a less robust feedback will lead to a faster convergence in the iteration domain. This also indicates that there might exist an optimal feedback gain that can achieve the best transient response in terms of convergence. Thus a random choice of pole location has larger effect in the iteration domain performance of the combined system. Clearly, it is also a trade-off between the time domain performance and iteration domain performance. It can be seen in Fig.3 that the different feedback choices have different time domain performance and that a high gain feedback system cannot always improve the learning performance.

A good feedback controller can achieve a reasonably good performance in time domain. However, it may take more iterations for the ILC controller to converge. The improved transient performance in the iteration domain is always critical in industrial applications. It is therefore interesting to understand how these feedback and ILC interact with each other and how these interactions have to be considered in the design process.

III. MAIN RESULTS

From the closed loop system (11) we can show that

$$\begin{aligned}\dot{e}_{i+1}(t) =& \dot{e}_i(t) + C\left(\delta\dot{\mathbf{x}}_{i+1}(t) - \delta\dot{\mathbf{x}}_i(t)\right)\\ =& \dot{e}_i(t) + C(A-BK)\left(\delta\mathbf{x}_{i+1}(t) - \delta\mathbf{x}_i(t)\right)\\ &+ CB\left(u_{i+1}^{ff}(t) - u_i^{ff}(t)\right)\\ =& \left(1-\gamma CB\right)\dot{e}_i(t) + C(A-BK)\Delta\mathbf{x}_{i+1}(t) \quad (14)\end{aligned}$$

where $\Delta\mathbf{x}_{i+1}(t) = \delta\mathbf{x}_{i+1}(t) - \delta\mathbf{x}_i(t) = \mathbf{x}_i(t) - \mathbf{x}_{i+1}(t)$.

Remark 3: It is obvious that, for a given feedback control algorithm, the feedback gain does not affect the convergence condition. That is, the convergence of the D-type ILC is independent of the choice of the feedback gains. However, as shown in the motivating example, the choice of the feedback will affect the transient response of the tracking error in iteration domain. Next will discuss how to choose an optimal feedback gain such that an optimal transient response in iteration domain is obtained. ∘

From equation (14), it follows that

$$\dot{e}_{i+1}(t) = \rho\dot{e}_i(t) + d_i(t), \quad (15)$$

where $\rho = (1-\gamma CB)$ and $d_i(t) = C(A-BK)\Delta\mathbf{x}_{i+1}(t)$. When the convergence condition (12) is satisfied by selecting the value of γ, the ILC algorithm (10) can ensure the convergence. In the proof of the convergence, the term $d_i(t)$ is made sufficiently small by using Gronwall Lemma and the standard time-weighted norm or a λ norm.

It is noted that the term $d_i(t)$ reflects the influence of dynamic systems in the convergence of tracking error in iteration domain. In order to evaluate this influence, the upper bound of d_i over the finite time interval is estimated. First, the upper bound of $|\Delta\mathbf{x}_i(t)|$ is estimated.

$$\delta\mathbf{x}_i = e^{(A-BK)t}\delta\mathbf{x}(0) + \int_0^t e^{(A-BK)(t-\tau)}B\delta u_i^{ff}d\tau \quad (16)$$

As $\delta\mathbf{x}(0) = 0$ as per $i.i.c$, we can subsequently show from (16) that

$$\begin{aligned}\Delta\mathbf{x}_{i+1}(t) &= \int_0^t e^{(A-BK)(t-\tau)}B\left(\delta u_{i+1}^{ff} - \delta u_i^{ff}\right)d\tau\\ |\Delta\mathbf{x}_{i+1}(t)| &\le \int_0^t \left|e^{(A-BK)(t-\tau)}\right|d\tau\,|B|\,|\gamma|\,\|\dot{e}_i\|_s. \quad (17)\end{aligned}$$

By applying Lemma 1, $e^{(A-BK)(t-\tau)} = T_\mu^{-1}e^{J_\mu(t-\tau)}T_\mu$, it follows that

$$\begin{aligned}&\int_0^t \left|e^{(A-BK)(t-\tau)}\right|d\tau\\ \le\ & \frac{1}{\left|\lambda_R(A-BK)+\frac{1}{\mu}\right|}|T_\mu|\left|T_\mu^{-1}\right|\left(1-e^{\left(\lambda_R(A-BK)+\frac{1}{\mu}\right)t}\right)\\ \le\ & \frac{1}{\left|\lambda_R(A-BK)+\frac{1}{\mu}\right|}|T_\mu|\left|T_\mu^{-1}\right|, \quad (18)\end{aligned}$$

as $\lambda_R(A-BK)+\frac{1}{\mu} < 0$.

We denote that $\phi(A-BK) := \frac{|A-BK|}{\left|\lambda_R(A-BK)+\frac{1}{\mu}\right|}|T_\mu|\left|T_\mu^{-1}\right|$. Taking the supremum norm for both sides of (14), by combining (17), (18) and the convergence condition (12), it follows that

$$|\dot{e}_{i+1}\|_s \le \rho\|\dot{e}_i\|_s + |\gamma|\,|C|\,|B|\,\phi(A-BK)\|\dot{e}_i\|_s. \quad (19)$$

Note that the feedback only affects $\phi(A-BK)$, which is also an iteration irrelevant function. Thus the term $\phi(A-BK)$ represents the dynamic influence in the iteration domain. If $\phi(A-BK)$ is minimized, the upper bound of the supremum norm in iteration domain will be smaller, leading to a better transient response.

However, it can be seen that the choice of μ also affects the performance. But μ is not a system parameter. It only provides a simplification used for estimating the upper bound of the matrix exponential. Intuitively, a smaller μ is preferred.

Thus it is desirable to design an optimal feedback gain, for example, feedback gain K in the problem formulation to minimize this term. The following optimization problem is thus formulated.

$$\min_{K\in\mathcal{S}}\phi(A-BK). \quad (20)$$

In general, finding an optimal solution for (20) is quite hard even if the plant model is completely known as it might be a non-convex optimization problem. The following property shows that in order to minimize $\phi(A-BK)$, the value of $\lambda_R(A-BK)$ can not be arbitrarily large.

Property 1: Let ϵ be an arbitrarily small positive constant and μ is a positive constant. Assume that $A-BK \in \mathcal{R}^{n\times n}$ is a Hurwitz and satisfies $\frac{1}{\mu|\lambda_R(A-BK)|} < 1$. Then minimum of $\frac{|A-BK|}{\left|\lambda_R(A-BK)+\frac{1}{\mu}\right|}$ is a function of $\lambda_R(A-BK)$. When $\lambda_R(A-BK) = \frac{1-\epsilon}{\epsilon\mu}$, this function reaches its minimal at $1-\epsilon$.

Proof: It is noted that $|A-BK| \geq |\lambda_R(A-BK)|$. In particular, if $\lambda_s(A-BK) = \lambda_R(A-BK), \forall s = 1,\ldots,n$, the equality holds. Since $A-BK$ is a Hurwitz, $\lambda_R(A-BK) \neq 0$. For any given $\mu > 0$, if $|A-BK| = |\lambda_R(A-BK)|$, $\frac{|A-BK|}{\left|\lambda_R(A-BK)+\frac{1}{\mu}\right|}$ reaches its minimum at $\frac{|\lambda_R(A-BK)|}{\left|\lambda_R(A-BK)+\frac{1}{\mu}\right|}$. Moreover, it has

$$\begin{aligned}\frac{|\lambda_R(A-BK)|}{\left|\lambda_R(A-BK)+\frac{1}{\mu}\right|} &= -\frac{|\lambda_R(A-BK)|}{\lambda_R(A-BK)+\frac{1}{\mu}} \\ &= -\frac{1}{\frac{\lambda_R(A-BK)+\frac{1}{\mu}}{|\lambda_R(A-BK)|}} \\ &= -\frac{1}{-1+\frac{1}{\mu|\lambda_R(A-BK)|}} \\ &= \frac{1}{1-\frac{1}{\mu|\lambda_R(A-BK)|}}\end{aligned}$$

where $\frac{1}{\mu|\lambda_R(A-BK)|} < 1$. A simple calculation shows that $\frac{|\lambda_R(A-BK)|}{\left|\lambda_R(A-BK)+\frac{1}{\mu}\right|} \leq 1-\epsilon$. This completes the proof. □

Remark 4: Property 1 shows that the dynamic influence in the iteration domain $\phi(A-BK)$ is related to $\lambda_R(A-BK)$. In order to minimize $\phi(A-BK)$, the value of $\lambda_R(A-BK)$ cannot be arbitrarily large. It is note that the value of $\lambda_R(A-BK)$ also represents the stability margin or robustness of the feedback control law in the time domain. Property 1 clearly shows that there is a design trade-off between robustness in time domain and the transient response in iteration domain as observed in [18], [19]. ◦

Remark 5: It is noted that the value of $\phi(A-BK)$ also depends on $|T_\mu|\left|T_\mu^{-1}\right|$. For a given μ, T_μ is related to the matrix $A-BK$. It can be shown that $|T_\mu|\left|T_\mu^{-1}\right|$ is related to locations of the poles of the matrix $A-BK$. Through a large number of simulations, it is observed that the optimal $A-BK$ always has poles that have the same real parts (possibly to be complex conjugate pairs), which are quite closed to the imaginary axis. Our future work will focus on providing a systematic design in selecting the eigenvalues of $A-BK$ with a rigorous proof. Intuitively, when there are complex conjugate pairs, the closed loop system exhibits oscillations, which will provide richer information to learn over iterations. ◦

Remark 6: This optimization problem is quite difficult to solve. Firstly, $\phi(A-BK)$ is a conservative estimate of the trajectories of the system in the closed-loop. Secondly, the standing assumption of ILC is that the plant of the system is not completely known. Thirdly, even if the nominal model of the plant is known, it is hard to show that the optimization problem (20) is convex. Last but not least, the parameters are constrained in set $\mathcal{S}$ to ensure the stability of the closed-loop in time domain. Thus in this paper, a simplified, but a practical algorithm is proposed to solve this problem.

Step 1 It is assumed that the nominal model of the system is obtained as (4). The nominal model is assumed to have enough accuracy.

Step 2 Under such a situation, an off-line optimization technique can be used to find the optimal feedback gain $\hat{K}^*$ either locally or globally.

Step 3 An on-line tuning algorithm is used to find $\hat{K}^*$ with appropriate choices of the cost function over iterations in order to handle the modeling uncertainties and disturbances.

It is possible to estimate the optimal feedback gains on-line in order to get the better transient response in the iteration domain without requiring the precise knowledge of the system. Moreover, the off-line optimal solutions from the nominal model can serve as a good initial guess to speed up the convergence. ◦

Remark 7: As noted, there is a design trade-off between the robustness in time domain and the transient response in iteration domain, a cost function can be designed as a linear combination of two costs (robustness and $\phi(A-BK)$). By choosing different weights for different applications, we can get a unified design for tuning feedback gains that can well balance these two performance indices. ◦

Remark 8: Our current problem formulation fixes the design of ILC while updating the feedback gain. It is possible to update the design of ILC in a similar way. Thus the problem formulation provides a unified framework to obtain an optimal performance in both iteration domain and time domain by selecting the parameters for ILC and feedback. Future work will extend this framework to more general systems such as nonlinear time-varying systems. ◦

IV. SIMULATION RESULTS

The simulation is performed to investigate the effect of optimal feedback gain to the transient response in iteration domain. A constrained offline optimisation using the inbuilt function *'fmincon'* in MATLAB software (MATLAB and Statistics Toolbox Release R2014b,The MathWorks Inc, MA) is performed.

The optimisation is performed for the system (13) for $\frac{1}{\mu} = 0.01$, resulted in optimal feedback gain, $K = [2.01, 0.93]$. The performance in the iteration domain for this feedback gain is then compared with other four cases used in the motivating example (Section II-C).

It can be observed that the optimized feedback parameters improve the transients in the iteration domain when compared with the choice of feedback gains considered in Section II-C. The supremum error converged to the minimum vale in least number of iterations. The time-domain performance in the first iteration when $u_1^{ff} = 0$ is shown in Fig.5. Even though the supremum error norm is larger in the first iteration compared to other cases,the current design allows a faster convergence of error.

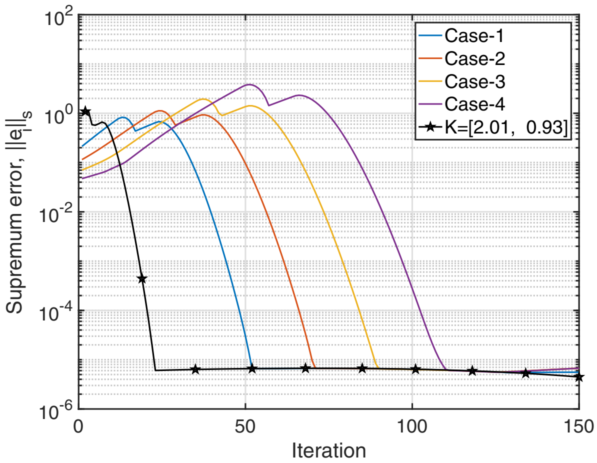

Fig. 4. Variation of supremum error for optimized results when compared with other choice of feedback gains

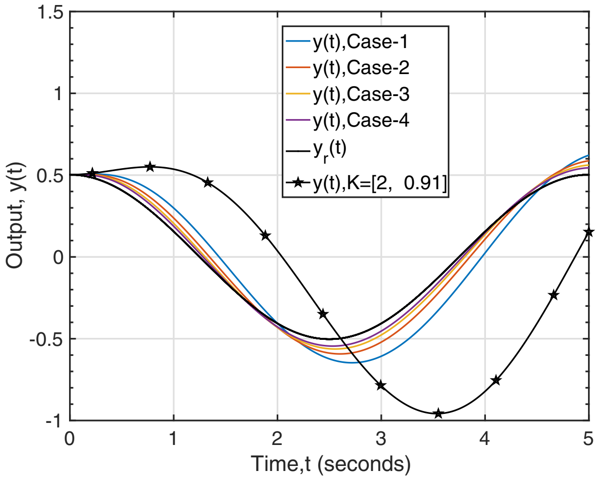

Fig. 5. Output response for when $u_1^{ff} = 0$

V. Conclusion and Future Work

A unified framework for an optimal design of a feedback controller to improve the transient performance in iterative domain is developed in this paper. A dynamic influence in the iteration domain is introduced which captures the effect of feedback controller in the convergence of error in the iteration domain. Optimal feedback control gains are selected to minimize this dynamic influence. It is observed that there is a clear trade-off between robustness in the time domain and transient performance in the iteration domain. The future work is to extend the framework to optimize the feedback controller parameters using an on-line gradient based optimization. Moreover, design optimal parameters for both ILC and feedback to balance the robustness in time domain and convergence in iteration domain will be explored.

Acknowledgement

This study is supported by the ARC Discovery Project DP160104018.

References

[1] R. W. Longman, "Iterative learning control and repetitive control for engineering practice," *International journal of control*, vol. 73, no. 10, pp. 930–954, 2000.
[2] K. L. Moore, *Iterative learning control for deterministic systems*. Springer Science & Business Media, 2012.
[3] C. Freeman, A.-M. Hughes, J. Burridge, P. Chappell, P. Lewin, and E. Rogers, "Iterative learning control of fes applied to the upper extremity for rehabilitation," *Control Engineering Practice*, vol. 17, no. 3, pp. 368–381, 2009.
[4] S.-H. Zhou, D. Oetomo, Y. Tan, E. Burdet, and I. Mareels, "Modeling individual human motor behavior through model reference iterative learning control," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 7, pp. 1892–1901, 2012.
[5] S.-H. Zhou, J. Fong, V. Crocher, Y. Tan, D. Oetomo, and I. Mareels, "Learning control in robot-assisted rehabilitation of motor skills–a review," *Journal of Control and Decision*, vol. 3, no. 1, pp. 19–43, 2016.
[6] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Systems*, vol. 26, no. 3, pp. 96–114, 2006.
[7] H.-S. Ahn, Y. Chen, and K. L. Moore, "Iterative learning control: brief survey and categorization," *IEEE Transactions on Systems Man and Cybernetics Part C Applications and Reviews*, vol. 37, no. 6, p. 1099, 2007.
[8] Y. Wang, F. Gao, and F. J. Doyle, "Survey on iterative learning control, repetitive control, and run-to-run control," *Journal of Process Control*, vol. 19, no. 10, pp. 1589–1600, 2009.
[9] C. Freeman and Y. Tan, "Iterative learning control and repetitive control," *International Journal of Control*, vol. 84, no. 7, pp. 1193–1195, 2011.
[10] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *Journal of Robotic systems*, vol. 1, no. 2, pp. 123–140, 1984.
[11] K. L. Moore, "Iterative learning control: An expository overview," in *Applied and computational control, signals, and circuits*. Springer, 1999, pp. 151–214.
[12] K. Lee, S. Bang, and K. Chang, "Feedback-assisted iterative learning control based on an inverse process model," *Journal of Process Control*, vol. 4, no. 2, pp. 77–89, 1994.
[13] N. Amann, D. H. Owens, E. Rogers, and A. Wahl, "An $H-\infty$ approach to linear iterative learning control design," *International Journal of Adaptive Control and Signal Processing*, vol. 10, no. 6, pp. 767–781, 1996.
[14] T.-Y. Doh, "Robust iterative learning control with current feedback for uncertain linear systems," *International Journal of Systems Science*, vol. 30, no. 1, pp. 39–47, 1999.
[15] T. Liu, X. Z. Wang, and J. Chen, "Robust pid based indirect-type iterative learning control for batch processes with time-varying uncertainties," *Journal of Process Control*, vol. 24, no. 12, pp. 95–106, 2014.
[16] N. Zsiga, S. van Dooren, P. Elbert, and C. H. Onder, "A new method for analysis and design of iterative learning control algorithms in the time-domain," *Control Engineering Practice*, vol. 57, pp. 39–49, 2016.
[17] K. Barton, S. Mishra, and E. Xargay, "Robust iterative learning control: L 1 adaptive feedback control in an ilc framework," in *American Control Conference (ACC), 2011*. IEEE, 2011, pp. 3663–3668.
[18] B. Altın and K. Barton, "Robust iterative learning for high precision motion control through l1 adaptive feedback," *Mechatronics*, vol. 24, no. 6, pp. 549–561, 2014.
[19] B. Altin and K. Barton, "$L1$ adaptive control in an iterative learning control framework: Stability, robustness and design trade-offs," in *American Control Conference (ACC), 2013*. IEEE, 2013, pp. 6697–6702.
[20] J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*. SIAM, 2000.
[21] G. Wang, Y. Tan, and I. Mareels, "Model-based predictive sampled-data control and its robustness," in *Control Conference (ASCC), 2015 10th Asian*. IEEE, 2015, pp. 1–6.
[22] J.-X. Xu and Y. Tan, *Linear and nonlinear iterative learning control*. Springer, 2003, vol. 291.
[23] D. H. Owens, "Iterative learning control-convergence using high gain feedback," in *Intelligent Control, 1992., Proceedings of the 1992 IEEE International Symposium on*. IEEE, 1992, pp. 455–457.