2008 American Control Conference
Westin Seattle Hotel, Seattle, Washington, USA
June 11-13, 2008

l_1-Optimal Robust Iterative Learning Controller Design

Kevin L. Moore *Senior Member, IEEE*
Division of Engineering, Colorado School of Mines
Golden, CO 80401, USA – kmoore@mines.edu

Mark H.A. Verwoerd
Hamilton Institute, National University of Ireland
Maynooth, Co. Kildare, Ireland – Mark.Verwoerd@nuim.ie

Abstract— **In this paper we consider the robust iterative learning control (ILC) design problem for SISO discrete-time linear plants subject to unknown, bounded disturbances. Using the supervector formulation of ILC, we apply a Youla parameterization to pose a MIMO l_1-optimal control problem. The problem is analyzed for three situations: (1) the case of arbitrary ILC controllers that use current iteration tracking error (CITE), but without explicit integrating action in iteration, (2) the case of arbitrary ILC controllers with CITE and with explicit integrating action in iteration, and (3) the case of ILC controllers without CITE but that force an integral action in iteration. Analysis of these cases shows that the best ILC controller for this problem when using a non-CITE ILC algorithm is a standard Arimoto-style update law, with the learning gain chosen to be the system inverse. Further, such an algorithm will always be worse than a CITE-based algorithm. It is also found that a trade-off exists between asymptotic tracking of reference trajectories and rejection of unknown-bounded disturbances and that ILC does not help alleviate this trade-off. Finally, the analysis reinforces results in the literature noting that for SISO discrete-time linear systems, first-order ILC algorithms can always do as well as higher-order ILC algorithms.**

Key Words: Iterative learning control, robust control, Youla parameterization, l_1-optimal control.

I. Introduction

The iterative learning control (ILC) paradigm for systems that operate in a repetitive fashion has attracted considerable attention and has a well-established research literature addressing both analysis and design (see the recent overview [1] for the basic ideas of the approach and the survey paper [2] for a detailed taxonomy of the literature). Though the basic questions of ILC have been understood under nominal assumptions about the plant to be controlled, there remain a number of open issues. One area of active research is related to robustness. Robustness can be considered from a variety of perspectives, including robustness with respect to variation in inputs and disturbances as well as robustness with respect to variation in process itself. For a summary of these issues, see [3].

In this paper we address a specific form of robustness that has not been considered in the ILC community: robustness with respect to disturbances that are unknown, but bounded, with a known bound. Specifically, we consider how to minimize the maximum value of the error given that we know the maximum value of the disturbance. In non-ILC control, this problem is known as the l_1-optimal control problem (in the discrete-time case, or the L_1 problem in the continuous-time case), because in the l_∞ signal space topology, the induced norm of the operator that maps the disturbance to the error is the l_1 norm [4]. While the l_1-optimal control problem is well-understood, we wish to investigate how the problem is impacted in an ILC scenario.

The paper is organized as follows. First we summarize the ILC framework in which we operate: a lifted representation we call the supervector approach. In this framework we then define our problem and apply the Youla parameterization to form a model matching problem with an l_1 optimality criteria. This formulation is done for three specific cases, distinguished by the presence or absence of current iteration feedback (CITE) and whether there is explicit use of an integrating action in iteration. Next, each scenario is analyzed to derive an l_1-optimal ILC controller and the controller's properties are discussed. The results indicate that in general ILC does not help improve the robustness of the system to l_∞ disturbances. The paper concludes with a summary of the results and a discussion of future research questions.

II. ILC Framework

A. The basic "supervector" approach

Let the SISO discrete-time plant $P(z)$ be given by

$$Y_k(z) \quad = \quad P(z)U_k(z) = (h_0 + h_1 z^{-1} + \cdots)U_k(z)$$

where the system is assumed (with some loss of generality) to have relative degree zero ($h_0 \neq 0$), z^{-1} is the standard delay operator with respect to time t, k denotes the iteration index, and the parameters h_i are the standard Markov parameters of the system $H(z)$. Per the normal ILC methodology [5], let the trial length be N and lift the time-domain signals to form the so-called supervectors:

$$\begin{aligned} U_k &= (u_k(0), u_k(1), \cdots, u_k(N)) \\ Y_k &= (y_k(0), y_k(1), \cdots, y_k(N)) \\ Y_\text{d} &= (y_\text{d}(0), y_\text{d}(1), \cdots, y_\text{d}(N)) \end{aligned}$$

from which we can write $Y_k = HU_k$, where H is a lower-triangular Toeplitz matrix of rank n whose elements are the

Markov parameters of the plant $P(z)$, given by:

$$H = \begin{bmatrix} h_0 & 0 & \cdots & 0 \\ h_1 & h_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h_{N-1} & h_{N-2} & \cdots & h_0 \end{bmatrix}$$

For each $t \in [0, N]$, let the (one-sided) w-transform $W(\cdot)$ be defined as

$$W(\{u_k(t)\}) \doteq \sum_{k=0}^{\infty} u_k(t) w^{-k} \tag{1}$$

We comment that the w-transform is similar to the standard z-transform, but it is operating from trial-to-trial, with time t fixed, as opposed to the standard z-transform operator, which operates from time step-to-time step, with k fixed. It should also be noted that to the authors' best knowledge, the first appearance of a transform operator operating from trial-to-trial was in [6]. The use of the transform operator has been developed in a number of citations. See [7], [8], [9], for example. However, again to our best knowledge, (1) is the first formal definition of the w-transform presented in the literature.

Note that using (1), we have

$$W(\{u_{k-1}(t)\}) = w^{-1} W(\{u_k(t)\}) + u_{-1}(t)$$

Thus, assuming $u_j = 0$ for all $j < 0$, we can write

$$W(\{u_{k-n}(t)\}) = w^{-n} W(\{u_k(t)\}).$$

In the sequel, we will apply the w-transform to our lifted vectors U_k, Y_k, and Y_d, to get vector w-transforms $U(w), Y(w)$, and $Y_\mathrm{d}(w)$, respectively. Thus, we may write the plant $Y_k = HU_k$ as $Y(w) = HU(w)$.

To proceed, consider the general form of a (higher-order) ILC algorithm:

$$\begin{aligned} U_{k+1} &= -\bar{D}_{n-1} U_k - \cdots - \bar{D}_0 U_{k-n+1} \\ &+ N_n E_{k+1} + N_{n-1} E_k + \cdots + N_0 E_{k-n+1} \end{aligned} \tag{2}$$

where, for all k, $E_k := Y_\mathrm{d} - Y_k$ denotes the tracking error. Observe that in this update law the "next input" is computed as a filtered sum of n "past inputs", "past errors", and "current errors", the latter via the current iteration tracking error (CITE) term $N_n E_{k+1}$[1] Taking the w-transform of both sides of this equation and combining terms gives

$$\bar{D}_c(w) U(w) = N_c(w) E(w)$$

where

$$\begin{aligned} \bar{D}_c(w) &= w^n + \bar{D}_{n-1} w^{n-1} + \cdots + \bar{D}_1 w + \bar{D}_0 \\ N_c(w) &= N_n w^n + N_{n-1} w^{n-1} + \cdots + N_1 w + N_0 \end{aligned}$$

[1] Standard ILC convention refers to the "order" of an ILC algorithm as the number of *past* trials that are used in computing the next input. Thus an algorithm such as $U_{k+1} = U_k + \Gamma E_k$ is a first-order ILC algorithm, while $U_{k+1} = U_k + \Gamma_1 E_{k-1} + \Gamma_2 E_{k-2}$ is considered a third-order algorithm (three past trials, $k, k-1$, and $k-2$ are used to compute the input on the next trial, $k+1$). Though the presence of a CITE term may produce some ambiguity, we will define order in terms of the maximum number of *past* trials used, either past errors or past inputs. Thus, the algorithm in (2) is an n-th order ILC update rule.

which can also be written in a matrix fraction as $U(w) = \bar{C}(w)E(w)$ where $C(w) = \bar{D}_c^{-1}(w) N_c(w)$. Note that the invertibility of $D_c(w)$ is subject to the same conditions as typically found in the theory of multivariable matrix fraction descriptions.

A common special case of (2) uses the general (higher-order) ILC update law

$$\begin{aligned} U_{k+1} &= (I - D_{n-2}) U_k + (D_{n-2} - D_{n-3}) U_{k-1} \\ &\quad + \cdots + (D_1 - D_0) U_{k-n+2} + D_0 U_{k-n+1} \\ &\quad + N_n E_{k+1} + \cdots + N_1 E_{k-n+2} + N_0 E_{k-n+1} \end{aligned} \tag{3}$$

Taking the w-transform of this update law now yields $(w-1) \cdot I \cdot D_c(w) U(w) = N_c(w) E(w)$, where

$$\begin{aligned} D_c(w) &= w^{n-1} + D_{n-2} w^{n-2} + \cdots + D_1 w + D_0 \\ N_c(w) &= N_n w^{n-1} + N_{n-1} w^{n-1} + \cdots + N_1 w + N_0 \end{aligned}$$

which can also be written in a matrix fraction as

$$U(w) = \frac{I}{(w-1)} C(w) E(w)$$

where $C(w) = D_c^{-1}(w) N_c(w)$.

Figure 1 depicts the set of equations we have just developed based on (3) for an ILC update law represented by $(w-1)^{-1} C(w)$ (more generally, we can replace the two blocks containing $C(w)$ and $1/(w-1)$ with a single block containing $\bar{C}(w)$ if using the general ILC update law of (2)). From this figure it is clear that the repetition-domain closed-loop dynamics from Y_d to $Y(w)$ become either:

$$\bar{G}_{cl}(w) = H[\bar{D}_c(w) + N_c(w) H]^{-1} N_c(w)$$

for (2) or, for (3),

$$G_{cl}(w) = H[(w-1) D_c(w) + N_c(w) H]^{-1} N_c(w)$$

For the latter case, because we now have an integrator in the feedback loop (a discrete integrator, in the repetition domain), applying the final value theorem to G_{cl} shows that $E_k \to 0$ as long as the ILC algorithm converges (i.e., as long as G_{cl} is stable).

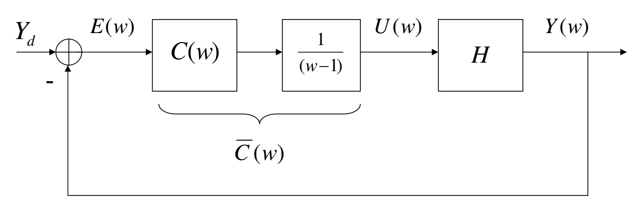

Fig. 1. Standard ILC setup in the supervector framework.

B. A generalized framework

As described in detail elsewhere [3], we can extend the development given above in a number of ways, by including:

1) Iteration-varying reference signals $Y_\mathrm{d}(w)$.
2) Iteration-varying noise signals $N(w)$.

3) Iteration-varying disturbances signals $D(w)$.
4) Iteration-varying nominal plant models $H(w)$.
5) Iteration-varying plant model uncertainty, $\Delta H(w)$.
6) Separation of the control action into current cycle feedback C_{CITE} and ILC update C_{ILC}.

Figure 2 depicts the complete picture. Note that in this figure we have assumed that integrating action (in the iteration domain) is used in the control law. Further, the diagram shows the current iteration feedback (CITE) separated from the ILC update. Both of these effects can be absorbed into a single controller denoted by $\bar{C}(w)$. In particular, note that the use of CITE has been incorporated into our algorithm by the term $N_n E_{k+1}$ in both (2) and (3). This simply means that with respect to our iteration-domain feedback system, the controller now has relative degree zero rather than the relative degree one controller that results when only previous cycle feedback is used.

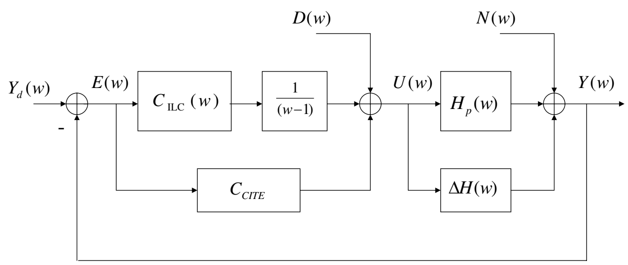

Fig. 2. More general ILC framework.

In the remainder of this paper we consider the ILC design problem for the case when the plant is subjected to an iteration-varying, unknown, and bounded disturbance $D(w)$. We will assume that there is no noise, that there is no iteration-variation for the plant or the reference signal, and that there is no uncertainty associated with the nominal model. With these assumptions and combining any integrating action or current cycle feedback into the controller, the general block diagram for our problem is given by Fig. 3, where we have dropped the overbar notation on the controller.

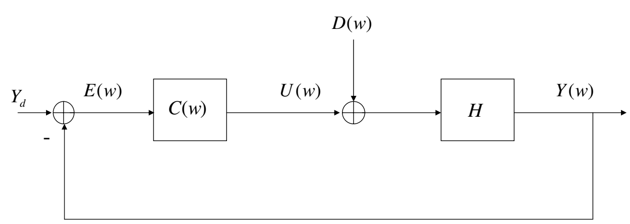

Fig. 3. General l_1 problem framework.

III. YOULA PARAMETERIZATION

Consider Figure 3, where again we note that the plant and reference signal are iteration-invariant, there is no noise, and we might have integrating action in the controller and we might have current cycle feedback (CITE). The signal $D(w)$ is assumed to be iteration-varying, unknown, and bounded. Our goal is to design a controller $C(w)$ to minimize the effect of the worst possible disturbance on the output. That is, if we denote the map from the disturbance to the output as $T_{YD}(w)$, our problem is[2]

$$\min_{C(w)} \|T_{YD}(w)\|_{l_1}$$

Our approach is to use the Youla parameterization, as suggested in [9] to convert this to a model matching problem and then solve the model matching problem with an l_1 optimality criteria.

As described in many places, e.g., [4], the Youla parameterization provides a characterization of all stabilizing controllers for a given system. Recall that $H(w) = ND^{-1} = \bar{D}^{-1}\bar{N}$ is a doubly coprime factorization if there exist stable, rational, coprime matrices $X, Y, \bar{X}$, and $\bar{Y}$ such that the Bezout equation

$$\begin{bmatrix} X & Y \\ -\bar{N} & \bar{D} \end{bmatrix} \begin{bmatrix} D & -\bar{Y} \\ N & \bar{X} \end{bmatrix} = I$$

is satisfied. Under these hypotheses the control law $U = -C(w)Y$ stabilizes the system, where

$$C(w) = (\bar{Y} + DQ)(\bar{X} - NQ)^{-1} = (X - Q\bar{N})^{-1}(Y + Q\bar{D})$$

and $Q(w)$ is any stable, rational matrix such that $\bar{X} - NQ \neq 0$ and $X - Q\bar{N} \neq 0$.

Because many ILC algorithms do no have a pure integrator in iteration, we consider three separate cases: not forcing an integrating action in iteration, forcing an integrating action in iteration while viewing the integrator as part of the controller, and forcing an integrating action in iteration while viewing the integrator as part of the plant. Note that the first two admit the possibility of CITE while the last case does not.

A. *Case 1 – No integrating action (in iteration)*

In this case we apply the Youla parameterization directly to Fig. 3, resulting in

- A possible non-zero error from Y_{d} to E, due to the fact that the l_1 optimization solution does not necessarily produce a controller with an integrating action. We remind the reader that the l_1 optimization problem is about minimizing the maximum error, which may amount to trading off good transient behavior against small asymptotic error.
- An arbitrary controller that will generically include a current cycle feedback component.
- A stable plant H used when carrying out the Youla parameterization. The resulting system equation to be used for design is $Y(w) = HU(w) + HD(w)$.
- A controller given by (for any stable, rational matrix $Q(w)$ such that $I - HQ(w) \neq 0$ and is biproper)

$$C(w) = Q(w)(I - HQ(w))^{-1}$$

[2]Because this is a unity feedback system, we can equivalently work with the influence of the disturbance on the output or on the error. In this paper we choose to consider the output.

- A closed-loop map from the disturbance to the output defined by

$$T_{YD}(w) \quad = \quad H - HQ(w)H \qquad (4)$$

B. Case 2 – Including an integrating action (in iteration) as part of the controller

To get zero steady-state error we need to include integrating action in the open-loop transfer function. We can do this directly by following the results in [10], which gives a parameterization of all stabilizing integrating controllers. To do this we apply the Youla parameterization to Fig. 4, resulting in

- A zero steady-state error from Y_d to E.
- An integrating controller that will generically include a current cycle feedback component.
- A stable plant H used when carrying out the Youla parameterization. The resulting system equation to be used for design is $Y(w) = HU(w) + HD(w)$.
- A controller given by (for any stable, rational matrix $Q(w)$ such that $I - Q(w)H \neq 0$ and is biproper, and any stabilizing ILC gain Γ[3])

$$C(w) = (I - Q(w)H)^{-1}(Q(w) + \frac{1}{w-1}\Gamma)$$

which, if we use $\Gamma = H^{-1}$, becomes[4]

$$C(w) = (I - Q(w)H)^{-1}(Q(w) + \frac{1}{w-1}H^{-1})$$

- A closed-loop map from the disturbance to the output defined by

$$T_{YD}(w) \quad = \quad \frac{w-1}{w}H - \frac{w-1}{w}HQ(w)H \qquad (5)$$

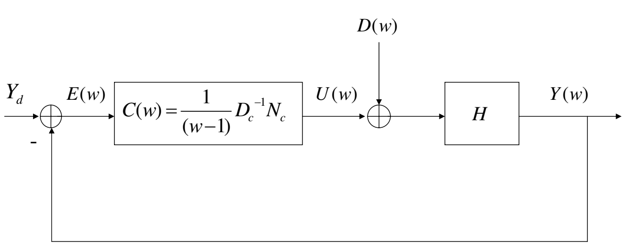

Fig. 4. l_1 problem framework using an integrating controller.

C. Case 3 – Including an integrating action (in iteration) as part of the plant

Another approach to getting the desired zero steady-state error is to define the "plant" to include the integrator that will actually be implemented in the controller and then design the remaining part of the controller to compensate for the resulting integrating plant. This approach, depicted in Fig. 5, results in

- A zero steady-state error from Y_d to E.
- An integrating controller that will not include a current cycle feedback component.
- An unstable plant $H/(w-1)$ used when carrying out the Youla parameterization. The resulting system equation to be used for design is $Y(w) = \frac{1}{w-1}HU(w) + HD(w)$.
- A controller given by (for any stable, rational matrix $Q(w)$ such that $I - HQ(w) \neq 0$)[5]

$$C(w) \quad = \quad ((1-\alpha)H^{-1} + \frac{(w-1)}{(w-\alpha)}Q(w)) \cdot (I - \frac{1}{(w-\alpha)}HQ(w))^{-1}$$

which, for $\alpha = 0$, is

$$C(w) = (wH^{-1} + (w-1)Q(w))(wI - HQ(w))^{-1}$$

- A closed-loop map from the disturbance to the output defined by (again, for $\alpha = 0$)

$$T_{YD}(w) \quad = \quad \frac{(w-1)}{w}H - \frac{(w-1)}{w^2}HQ(w)H \quad (6)$$

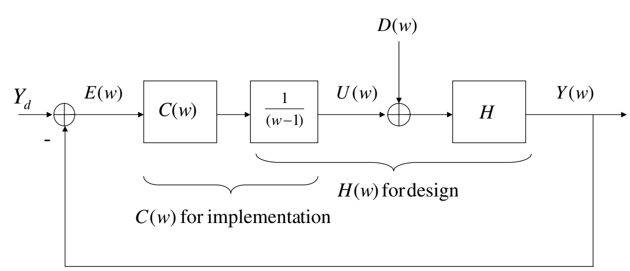

Fig. 5. l_1 problem framework using an integrating plant.

IV. l_1 OPTIMIZATION

Based on the analysis in the previous section, we now consider the problem:

$$\min_{Q(w)} \|T_{YD}\|_{l_1}$$

[3]When forcing an integrating action in iteration, we are effectively using a controller $U_{k+1} = U_k + \Gamma E_k$, or $U(w) = \frac{1}{(w-1)}\Gamma E(w)$. Thus, as per [10], the condition on Γ is that the controller $C(w) = \frac{1}{(w-1)}\Gamma$ should stabilize H.

[4]Note that because the Youla parameterization gives all stabilizing controllers for any Γ that stabilizes the plant, there is no loss of generality due to any specific choice of Γ.

[5]To develop the Youla parameterization for this case, we pick α, $|\alpha| < 1$, and define the plant $H(w) = ND^{-1} = \bar{D}^{-1}\bar{N}$, where $D = \bar{D} = \frac{(w-1)}{(w-\alpha)}I$, and $N = \bar{N} = \frac{1}{(w-\alpha)}H$. One solution to the corresponding Bezout equation $X(w-1) + YH = (w-\alpha)I$ is given by $X = I$, $Y = (1-\alpha)H^{-1}$, resulting in the controller indicated. Note that the choice of the Youla parameters is immaterial with respect to the synthesis problem considered here.

The three different parameterizations, (4), (5), and (6), repeated here and shown with a subscript for convenience, are:

$$\begin{aligned} T_{YD_1}(w) &= H - HQ(w)H \\ T_{YD_2}(w) &= \frac{w-1}{w}H - \frac{w-1}{w}HQ(w)H \\ T_{YD_3}(w) &= \frac{(w-1)}{w}H - \frac{(w-1)}{w^2}HQ(w)H \end{aligned}$$

Note that by using the Youla parameterization, we have ensured closed-loop stability. Our interest then is to minimize the effect of an unknown but bounded disturbance. It would initially be tempting to think that we can solve the problem for all three cases by simply letting $Q(w) = H^{-1}$, which would give $T_{YD}(w) = 0$. However, in this case the Youla parameterization requirement that $(I - Q(w)H)$ is biproper would be violated.

A. Case 1 – No integrating action (in iteration)

Though we might turn to the more sophisticated machinery of [4], it turns out that this is not necessary initially. Without loss of generality, let $Q(w)$ be given as

$$Q(w) = Q_0 + Q_1 w^{-1} + Q_2 w^{-2} + \dots$$

. Then $\|H - HQ(w)H\|_{l_1}$ becomes

$$\begin{aligned} \|T_{YD_1}(w)\|_{l_1} &= \|H - H(Q_0 + Q_1 w^{-1} + \dots)H\|_{l_1} \\ &= \|H - HQ_0H\|_{l_1} + \sum_{i=1}^{\infty} \|HQ_iH\|_{l_1} \end{aligned}$$

Clearly, to minimize this, we want to make all the terms in the sum zero or as close to zero as possible without violating the constraint $(I - Q(w)H)$ biproper (the only way it can be zero is if $Q(w) = H^{-1}$, clearly not a viable choice). There is more than one way to do this (there is not a unique solution), so we chose the one that will give the lowest-order controller. Specifically, let $Q_i = 0$ for $i > 0$ and set

$$Q_0 = H^{-1} - \gamma H^{-2}$$

where $\gamma > 0$. Then we get $\|T_{YD_1}(w)\|_{l_1} = \gamma$, which can be made arbitrarily small[6]. It is also interesting to compute the resulting controller, which becomes:

$$C(w) = \frac{1}{\gamma}I - H^{-1}$$

and the resulting output signal, including the reference:

$$Y = (I - \gamma H^{-1})Y_{\mathrm{d}} + \gamma D(w)$$

Thus we see that this controller allows us make the error arbitrarily small, at the cost of a using a high-gain controller (the $\frac{1}{\gamma}$ in the controller equation). Also note that the controller is independent of iteration. That is, in a worst-case sense, ILC does nothing to improve the rejection of an unknown, bounded disturbance.

[6] As pointed out by a reviewer, making $\|T_{YD_1}(w)\|_{l_1}$ arbitrarily small means that in the limit $(I - Q(w)H)$ will violate the requirement to be biproper. However, the biproper condition is satisfied for any finite γ. In this sense the result is no less meaningful than any typical high-gain feedback result in adaptive control.

Note that alternate choices of $Q(w)$ are possible. When we have forced the higher-order terms in $Q(w)$ to zero, the alternate choices of Q_0 seem to always produce a high gain controller. It is interesting to ask if we can get the same performance with a lower-gain controller if we allow higher-order terms.

B. Case 2 – Including an integrating action (in iteration) as part of the controller

Next, consider the problem

$$\min_{Q(w)} \|T_{YD_2}(w)\|_{l_1}$$

where

$$T_{YD_2}(w) = \frac{w-1}{w}H - \frac{w-1}{w}HQ(w)H$$

In this case it is tempting to use

$$Q(w) = \frac{1}{(w-1)}(Q_0 w - H^{-1})$$

where Q_0 is the solution given in Case 1 above. Such a choice makes $T_{YD_2}(w) = T_{YD_1}(w) = \gamma I$, the same solution as in Case 1. Unfortunately, this is not an allowable $Q(w)$, because it is not stable (due to the integrator that appears in $Q(w)$) and indeed, a computation of the resulting controller gives the same controller as in Case 1, $C(w) = (1/\gamma)I - H^{-1}$, which clearly does not contain an integrating action.

To proceed we cannot directly apply the standard l_1-optimization machinery from [4], because our problem introduces interpolation constraints on the stability boundary (zeros at $w = 1$). However, a result from [4] does tell us that even if we cannot obtain the optimal, we can get arbitrarily close with an FIR solution (that is, by making $T_{YD_2}(w)$ FIR). Exploiting this result, let us use the exact same $Q(w)$ as in Case 1:

$$Q(w) = Q_0 = H^{-1} - \gamma H^{-2}$$

which results in $\|T_{YD_2}(w)\|_{l_1} = \|\gamma \frac{(w-1)}{w} I\|_{l_1} = 2\gamma$, with a resulting controller

$$C(w) = \frac{1}{(w-1)}\left[(\frac{1}{\gamma}I - H^{-1})w + H^{-1}\right]$$

and closed-loop output signal

$$Y = (I - \gamma\frac{(w-1)}{w}H^{-1})Y_{\mathrm{d}} + \gamma\frac{(w-1)}{w}D(w)$$

Here we clearly see that the integrator makes the steady-state error zero between the desired (constant) signal Y_{d} and the output Y(to see this, let Y_{d} be a step and then apply the final value theorem), while allowing the error due to the disturbance to be pushed arbitrarily small. Of course, we again see that the controller has a high-gain characteristic as in Case 1. Also, as in Case 1, the ILC action does nothing, in a worst-case sense, to improve the performance relative to the disturbance. But, by adding the ILC feature, we gain the benefit of making the error with respect to the reference go to zero. We can also observe that

1) The price we pay for forcing an integrator in the system (Case 2) to get asymptotic convergence (in iteration) with respect to the reference signal is a higher gain from the l_∞ disturbance to the output than when the integrator is not used (Case 1).
2) The best possible l_∞ disturbance rejection for the case of an integrator with the possibility of CITE (Case 2) is the same as that for CITE alone (Case 1).

C. Case 3 – Including an integrating action (in iteration) as part of the plant

Finally we consider the problem

$$\min_{Q(w)} \|T_{YD_3}(w)\|_{l_1}$$

where

$$T_{YD_3}(w) = \frac{w-1}{w}H - \frac{w-1}{w^2}HQ(w)H$$

As in Case 2, we cannot choose a $Q(w)$ that contains an integrator and because of the interpolation constraints, we seek to make $T_{YD_3}(w)$ FIR. Assuming

$$Q(w) = Q_0 + Q_1 w^{-1} + Q_2 w^{-2} + \dots$$

gives

$$\begin{aligned} T_{YD_3}(w) &= H - (H + HQ_0H)w^{-1} \\ &\quad - H(Q_1 - Q_0)Hw^{-2} - \dots \end{aligned}$$

so that

$$\begin{aligned} \|T_{YD_3}(w)\|_{l_1} &= \|H\|_{l_1} + \|H + HQ_0H\|_{l_1} \\ &\quad + \|H(Q_1 - Q_0)H\|_{l_1} + \dots \\ &\geq 2\|H\|_{l_1} \end{aligned}$$

The minimum l_1 gain occurs when $Q(w) = 0$,[7] which results in a controller $C(w) = H^{-1}$ and a closed-loop response given by

$$Y = \frac{(w-1)}{w}Y_{\rm d} + \frac{(w-1)}{w}HD(w)$$

Thus, we see that when we use a non-CITE ILC algorithm the best we can do is use a standard Arimoto-style update law, with the learning gain chosen to be the system inverse. Further, such an algorithm will always be worse than a CITE-based algorithm. This means that ILC cannot help in the case of a system subject to unknown, bounded inputs.

V. Conclusion

In this paper we have studied the design of iterative learning controllers for plants subject to unknown but bounded disturbances. We have shown that when CITE ILC algorithms are used, the best case l_∞ disturbance attenuation is better than when non-CITE algorithms are used. Further, when CITE is employed, there is a trade-off between asymptotic convergence (due to the use of an integrating action in iteration) and l_∞ disturbance attenuation, with better disturbance rejection achieved when the controller does not use an explicit integrating action. However, in both the case of no integrating action and integrating action, the disturbance attenuation can be made arbitrarily small, but at the cost of a high gain controller. We also point out that in all three cases considered, the resulting ILC algorithm is always first order. This corresponds to many of the results presented in a special session devoted to higher-order ILC at the 2002 IFAC World Congress (see [11]). As a final comment, we note that in [12] and [13] we have taken a state-space based algebraic approach to the robust ILC problem when there are l_2 disturbances, which is equivalent to the problem:

$$\min_{C(w)} \|T_{YD}(w)\|_{H_\infty}$$

In our future research we will reconsider this problem using a model-matching approach.

[7] This is not the only choice of $Q(w)$ that achieves the minimum. For instance, it is also attained when $Q(w) = -H^{-1}$

References

[1] Douglas A. Bristow Marina Tharayil and Andrew G. Alleyne, "A survey of iterative learning control: A learning-based method for high-performance tracking control," *IEEE Control Systems Magazine*, vol. 26, no. 3, pp. 96–114, 2006.

[2] Ahn H.-S. Chen Y.-Q. and Moore K.L., "Iterative learning control: Brief survey and categorization," *IEEE Transactions on Systems, Man, and Cybernetics, Part-C*, vol. 37, 2007.

[3] Ahn H.-S. Moore K.L. and Chen Y.-Q., *Iterative Learning Control: Robustness and Monotonic Convergence for Interval Systems*, Springer-Verlag, London, 2007.

[4] Dahleh M. and Diaz-Bobillo I. J., *Control of Uncertain Systems: A Linear Programming Approach*, Prentice Hall, New Jersey, 1995.

[5] K. L. Moore, "A matrix fraction appraoch to higher-order iterative learning control: 2-D dynamics through repetition-domain filtering," in *Proceedings of the Second International Workshop on Multidimensional (ND) Systems*, Czocha Castle, Poland, June 27 - 30 2000, pp. 99–104.

[6] Kevin L. Moore, "Multi-loop control approach to designing iterative learning controllers," in *Proceedings of the 37th IEEE Conference on Decision and Control*, Tampa, Florida, USA, 1998, pp. 666–671.

[7] J. J. Hatonen D. H. Owens and K. L. Moore, "An algebraic approach to iterative learning control," *International Journal of Control*, vol. 77, no. 1, pp. 45–54, 2004.

[8] D. H. Owens, E. Rogers, and K. L. Moore, "Analysis of linear iterative learning control schemes using repetitive process theory," *Asian Journal of Control, Special Issue on ILC*, vol. 4, no. 1, pp. 90–98, March 2002.

[9] Verwoerd M. Meinsma G and de Vries T., "On admissible pairs and equivalent feedbackyoula parameterization in iterative learning control," *Automatica*, vol. 42, no. 12, pp. 2079–2089, 2006.

[10] Gundes A. N. and Kabuli M. G., "Parameterization of stabilizing controllers with integral action," *IEEE Transaction on Automatic Control*, vol. 44, no. 1, pp. 116–119, 1999.

[11] K. L. Moore and YangQuan Chen, "On monotonic convergence of high order iterative learning update laws," in *Invited Session on* High-order Iterative Learning Control *at The 15-th IFAC Congress*, Barcelona, Spain, July 21-26 2002, IFAC.

[12] Moore K.L. Ahn H.-S. and Chen Y.-Q., "Iteration-domain h-infinity iterative learning controller design," *International Journal of Robust and Nonlinear Control*, accepted to appear 2008.

[13] Moore K.L. Ahn H.-S. and Chen Y.-Q., "Algebraic h-infinity design of higher-order iterative learning controllers," in *Proceedings 2005 IEEE International Symposium on Intelligent Control*, Cyprus, Greece, June 2005.