Proceedings of the 2005 IEEE
International Symposium on Intelligent Control
Limassol, Cyprus, June 27-29, 2005

Algebraic H_∞ Design of Higher-Order Iterative Learning Controllers

Kevin L. Moore[†], *Senior Member, IEEE*, YangQuan Chen[‡], *Senior Member, IEEE*, and Hyo-Sung Ahn[‡], *Student Member, IEEE*

[†]Research and Technology Development Center
Johns Hopkins University Applied Physics Laboratory
11100 Johns Hopkins Road, Laurel, MD 20723-6099
kevin.moore@jhuapl.edu

[‡]Center for Self-Organizing and Intelligent Systems (CSOIS)
Dept. of Electrical and Computer Engineering
4160 Old Main Hill, Utah State University, Logan, UT 84322-4160, USA
yqchen@ece.usu.edu,hyosung@cc.usu.edu

***Abstract*— This paper presents an H_∞-based design technique for synthesis of higher-order iterative learning controllers (ILC) for plants subject to iteration-domain input/output disturbances and plant model uncertainty. Using a formulation of the higher-order ILC problem into a high-dimensional MIMO discrete-time system, it is shown how the addition of input/output disturbances and plant model uncertainty to the ILC problem can be cast into a standard H_∞ framework. An algebraic approach to solving the problem in this framework is presented, resulting in a sub-optimal controller that can achieve both stability and robust performance. The key observation is that H_∞ synthesis can be used for higher-order ILC design to achieve reliable performance in the presence of iteration-varying external disturbances and model uncertainty.**

***Index Terms*— Iterative learning control, higher-order ILC, H_∞ sub-optimal control, iteration varying uncertainty.**

I. Introduction

Iterative learning control (ILC [1-4]), a technique that attempts to refine the performance of systems that repeat their operation over and over from the same initial conditions, is fundamentally a two-dimensional process, with evolution along both a finite time axis (denoted by t) and an infinite iteration axis (denoted by k) [5]. In most ILC design methodologies, the learning gain or learning gain matrix is designed using the nominal plant model. Well-established convergence analysis results show that generally, even with a badly-modeled nominal plant, the required performance can be effectively achieved as the trial number (iteration count) increases. Further, in the ILC literature a number of results are available that consider robustness issues, including H_∞ ILC design from the time-axis perspective [6], [7], [8], stochastic ILC design [9], and disturbance rejection [10], [11], [12], [13]. However, for the most part, existing work has focused on ILC design for performance improvement, with the assumption that the plant is iteration-invariant and with external disturbances treated along the time axis. That is, the primary focus is on robustness defined and modeled along the time axis. To date, iteration axis robustness has not been treated in a systematic way (one exception to this, in a somewhat different framework, is the work on multi-pass systems [14], [15]). In this paper, a new framework is suggested for robust ILC design assuming both test-varying model uncertainty and iteration-varying external disturbances. Our work is developed based on the so-called super-vector framework [16], [17], [18], [19], a lifting technique for converting the two-dimensional ILC problem into a one-dimensional multi-input, multi-output (MIMO) problem. Using the super-vector approach we can easily incorporate iteration-varying disturbances and the system can be analyzed using discrete (iteration axis) frequency domain techniques.

Corresponding author: Prof. YangQuan Chen. E-mail: yqchen@ece.usu.edu; Tel. 01-435-7970148; Fax: 01-435-7973054. URL: http://www.csois.usu.edu/ilc.

The paper is organized as follows. In Section II, we summarize the formulation of the higher-order ILC problem into a high-dimensional MIMO discrete-time system and it is shown how the addition of input/output disturbances and plant model uncertainty to the ILC problem can be cast into a standard H_∞ framework. It is supposed that the disturbance and uncertainty could be any kind signal, but only their l_2 boundary is assumed known, which is a general requirement in H_∞ design. In Section III, an algebraic H_∞ solution to the design of a higher-order ILC algorithm is presented, resulting in a sub-optimal controller that can achieve both stability and robust performance. Simulation results are given in Section IV. We further discuss some related issues of H_∞ ILC and on-going work in Section V.

II. Problem Formulation

Following [16], let the plant $H(z)$ be given by

$$\begin{aligned} Y(z) &= H(z)U(z) = (h_1 z^{-1} + h_2 z^{-(2)} \\ &\quad + h_3 z^{-(3)} + \cdots)U(z), \end{aligned} \tag{1}$$

where the system is assumed, with no loss of generality, to have relative degree one, z^{-1} is the standard delay operator along the time axis, and the parameters h_i are the standard Markov parameters of the system $H(z)$. We define:

$$\begin{aligned} U_k &= (u_k(0), u_k(1), \cdots, u_k(N-1)), \\ Y_k &= (y_k(1), y_k(2), \cdots, y_k(N)), \\ Y_d &= (y_d(1), y_d(2), \cdots, y_d(N)), \end{aligned} \tag{2}$$

and 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 $H(z)$, given by:

$$H = \begin{bmatrix} h_1 & 0 & \cdots & 0 \\ h_2 & h_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h_N & h_{N-1} & \cdots & h_1 \end{bmatrix}.$$

Next, introduce a delay operator along the iteration axis, w, with the property that

$$w^{-1}u_k(t) = u_{k-1}(t).$$

We call this the "w-transform." It operates from trial-to-trial, with t fixed, as opposed to the standard z-transform operator which operates from time step-to-time step, with k fixed. Thus, we may write $Y_k = HU_k$ as $Y_k(w) = HU_k(w)$. Using this, we can formulate the ILC design problem as follows. Consider a general higher-order ILC algorithm of the form:

$$\begin{aligned} U_{k+1} &= (I - D_{n-1})U_k + (D_{n-1} - D_{n-2})U_{k-1} + \cdots \\ &+ (D_2 - D_1)U_{k-n+2} + (D_1 - D_0)U_{k-n+1} + D_0 U_{k-n} \\ &+ N_n E_k + N_{n-1}E_{k-1} + \cdots + N_1 E_{k-n+1} + N_0 E_{k-n}. \end{aligned}$$

If we take the "w-transform" of this ILC update equation, combine terms, and simplify we get:

$$(w-1)D_c(w)U(w) = N_c(w)E(w),$$

where

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

This can also be written in a matrix fraction as

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

where

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

For this update law the repetition-domain closed-loop dynamics become:

$$\begin{aligned} G_{cl}(w) &= H(I + \frac{I}{(w-1)}C(w)H)^{-1}\frac{I}{(w-1)}C(w), \\ &= H[(w-1)D_c(w) + N_c(w)H]^{-1}N_c(w). \end{aligned}$$

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} gives $E_k \to 0$ as long as the ILC algorithm converges (i.e., as long as G_{cl} is stable).

Fig. 1(a) depicts the general higher-order ILC problem as a MIMO control problem with the plant H. This figure highlights the fact that the ILC process (1) inherently is a relative degree one process; and (2) should have an integrator in order to converge to zero steady-state error. However, it can also be noted that the controller $C(w)$ has relative degree zero. This makes it convenient to consider the reformulation shown in Fig. 1(b). Here:

1) The integrator has been grouped with the plant, so that we now define a new plant

$$H_p(w) = (w-1)^{-1}H.$$

2) We suppose that H is subject to a perturbation such as $H = H_0 + \Delta H(w)$ where $\Delta H(w)$ represents iteration varying uncertainty in the plant model.
3) The plant is disturbed by a plant input disturbance d_I and a plant output disturbance d_o.

These disturbances and plant perturbation models lead to a standard H_∞ problem. Specifically, the design problem for the uncertain ILC system can be formulated as:

Problem: Given $H_p(w) = (w-1)^{-1}H$ and $Y_d(w)$, find $C(w)$ in Fig. 1(b) such that $\|E_k\|_2$ is minimum from the l_2-bounded disturbances d_I and d_o and the closed-loop system is stable over all $H = H_0 + \Delta H(w)$, with $\|\Delta H(w)\|_\infty < \epsilon_H$.

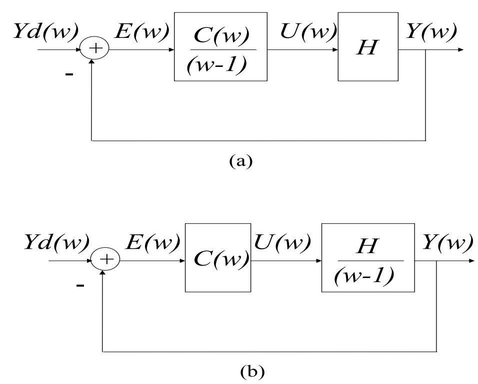

Fig. 1. Super-vector ILC.

Definition 2.1: In super-vector ILC, the l_2 norm is defined along the iteration axis (i.e., on the w domain). To distinguish this concept from the discrete time domain, the iteration domain l_2 norm is written as: $\|\cdot\|_{2^w}$ and denoted l_{2^w}. For example, we replace $\|E_k\|_2$ in the problem statement by $\|E_k\|_{2^w}$. In the same way, the iteration domain H_∞ norm is written as: $\|\cdot\|_{\infty^w}$

Notice that the problem set-up above indicates that the high order ILC system can be synthesized from the H_∞ framework. That is, the minimization problem of $\|E(w)\|_{2^w}$ is translated as the minimization problem of

$$\|T_{EW}\|_{\infty^w} = \sup_{W_k \in l_{2^w}} \frac{\|E_k\|_{2^w}}{\|W_k\|_{2^w}}$$

such that

$$\|T_{EW}\|_{\infty^w} < \gamma \tag{3}$$

where $W = [d_I, d_o]^T$, $\|E_k\|_{2^w} = \sum_{k=0}^{k=\infty} E(k)^T E(k)$ and $\|W_k\|_{2^w} = \sum_{k=0}^{k=\infty} W(k)^T W(k)$ [1]. When γ is not fixed, this is an optimal H_∞ ILC problem and when γ is fixed this a the sub-optimal H_∞ ILC.

Remark 2.1: In ILC, the minimization of T_{EW} means the reduction of the H_∞ gain of the transfer matrices from W_k to E_k. In other words, the reference input Y_d is not counted in this performance problem. So, minimization of T_{EW} does not guarantee the minimization of $E_k = Y_d - Y_k$. Instead, the sensitivities of d_I and d_o to E_k are reduced by minimization of $\|T_{EW}\|_{\infty^w}$. The minimization of $E_k = Y_d - Y_k$ is ensured by the presence of the integrator in the loop gain and by adequate solution of the robust stability problem.

In our work, we try to design the ILC controller $C(w)$ with fixed $\gamma = 1$. Further notice that:

1) $H_p(w)$ has what is called a structured perturbation, because we have

$$H_p(w) = (w-1)^{-1}H = \frac{I}{(w-1)}\left(H_0 + \Delta H(w)\right),$$

not

$$H_p = H_{p_0} + \Delta H(w).$$

[1]In this paper, E_k and $E(k)$ have the same meaning. Both represent the iteration trial number. For convenience, we use the both symbols.

That is, there is no modelling uncertainty associated with the integrator, as it is actually due to the controller, not the plant.

2) Fig. 1(b), with (3), is the classic H_∞ robust control problem. Basically, the point is to formulate and solve any of the robust stability and robust performance problems with H_∞ existence solution.

III. Algebraic H_∞ Approach in ILC

In this section we consider the robust design problem for external disturbances and for model uncertainty separately.

A. Iteration-Varying Disturbances

Fig. 2 shows the block diagram of the general H_∞ problem, which can be written in state-space form as:

$$
\begin{aligned}
x_{k+1} &= Ax_k + B_1 w_k + B_2 u_k && (4)\\
z_k &= C_1 x_k + D_{11} w_k + D_{12} u_k && (5)\\
y_k &= C_2 x_k + D_{21} w_k + D_{22} u_k && (6)
\end{aligned}
$$

where $z_k = [z_k^1, z_k^2]^T$ is the performance outputs, y_k is the observation output to be used in output feedback control, and $w_k = [d_I, d_o]^T$ are the exogenous inputs (disturbances in plant input and plant output). Generally, it is convenient to assume that $D_{11} = 0$ (or, D_{11} should be very small to hold the existence condition) and $D_{22} = 0$. In Fig. 2, W_1 and W_2 are penalty weighting matrices, and W_i and W_o are disturbance generating functions. In our ILC problem, W_1 and W_2 are identity matrices, and $W_i = W_o = \alpha I$.

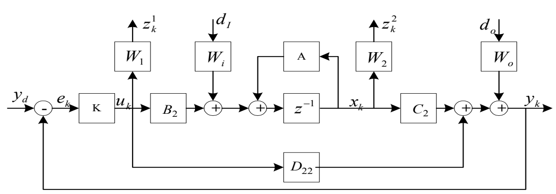

Fig. 2. Typical discrete H_∞ diagram.

For a solution K to exist in Fig. 2, we need the following assumptions:

A1: (A, B_2) is stabilizable and (C_2, A) is detectable.

A2: D_{12} is full column rank and D_{21} is full row rank.

The controller existence condition such that $\|T_{zw}\|_\infty < 1$ while stabilizing the system is given in many references. See [20], for example.

Now, we show that the higher-order ILC problem can be cast into the standard H_∞ framework of Fig. 2. From Fig. 1 (b), with $\Delta H(w) = 0$, the higher-order ILC system can be written as:

$$
\begin{aligned}
U_{k+1} &= U_k + V_k + d_I\\
Y &= HU_k + d_o && (7)
\end{aligned}
$$

where we have defined $V_k(w) = C(w)E(w)$.

To proceed, we will reformulate (7) into a state space form corresponding to (4), (5), and (6), from which we can re-draw Fig. 1(b) into the form of Fig. 2. The outcome of this will be Fig. 3. When done, in Fig. 3, the performance penalty will be selected as $Z_k = [z_k^1, z_k^2]^T = [E_k, D_{12}V_k]^T$, where $V_k = C(w)E_k$, such that the input plant sensitivity matrix (from d_I to E_k) becomes:

$$S_I = H_p(w)(I + C(w)H_p(w))^{-1}$$

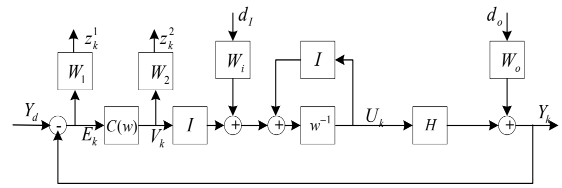

Fig. 3. ILC H_∞ diagram with plant input and output disturbances.

and the output sensitivity matrix becomes:

$$S_o = (I + C(w)H_p(w))^{-1}.$$

In this performance penalty, $D_{12}V_k$ is used to satisfy the full column rank requirement and also to minimize the control effort due to the external disturbance. Therefore, the H_∞ performance problem is to minimize two sensitivity matrices (i.e., S_I and S_o) and is to stabilize the ILC system (robust stability problem). These two goals define the robust ILC performance problem.

Continuing, the following state-space form of the ILC equations can be derived:

$$
\begin{aligned}
U_{k+1} &= IU_k + B_1 W_k + IV_k && (8)\\
Z_k &= [H, 0]^T U_k + D_{11} W_k + D_{12} V_k && (9)\\
Y_k &= HU_k + D_{21} W_k, && (10)
\end{aligned}
$$

where H is the Markov matrix, $W_k = [d_I, d_o]^T$, $B_1 = [\alpha I \ 0]$, $V_k = C(w)E_k$, and $D_{11} = \begin{bmatrix} 0 & \alpha I \\ 0 & 0 \end{bmatrix}$, $D_{21} = [0, \ \alpha I]^T$ (α is used to limit the disturbance intensity). Notice that there is an one-to-one mapping between equations (4,5,6) and (8,9,10) and also between Fig. 2 and Fig. 3. Hence, we can apply the general H_∞ framework to the higher-order super-vector ILC description, because (A, B_2)=(I, I) is stabilizable and (C_2, A)=(H, I) is detectable with any H. As in the standard H_∞ problem set-up, in ILC the following state-space formula can be suggested:

$$A = I; \ B_1 = [\alpha I \ 0]; \ B_2 = I; \ C_1 = [H, \ 0]^T; \ C_2 = H;$$

$$D_{11} = \begin{bmatrix} 0 & \alpha I \\ 0 & 0 \end{bmatrix}; \ D_{12} = [0, \ I]^T; \ D_{21} = [0, \ \alpha I]; \ D_{22} = 0$$

With this set-up, the H_∞-based design of the super-vector ILC controller becomes a typical H_∞ synthesis problem.

Now, for an analytical solution, the following definitions are provided. Let

$$
\begin{aligned}
\Theta_1 &:= \begin{bmatrix} I - \alpha^2 \overline{X} & 0 \\ 0 & I - \alpha^2 I \end{bmatrix}.\\
\widetilde{B_1} &:= [\alpha \overline{X}, 0]^T; \ \widetilde{B_2} := \overline{X} + H; \ \widetilde{B_3} := [\alpha \overline{X}, \alpha I]^T\\
\Theta_2 &:= I + \overline{X} + \alpha^2 \overline{X}(I - \alpha^2 \overline{X})^{-1} \overline{X} + \frac{\alpha^2}{1 - \alpha^2} I\\
\Theta_3 &:= H + \overline{X} + \alpha^2 \overline{X}(I - \alpha^2 X)^{-1} \overline{X}\\
\overline{A} &:= I + \alpha^2 (I - \alpha^2 \overline{X})^{-1} \overline{X}\\
\overline{B}_1 &:= [\alpha (I - \alpha^2 \overline{X})^{-\frac{1}{2}}, 0]\\
\overline{B}_2 &:= I + \alpha^2 (I - \alpha^2 \overline{X})^{-1} \overline{X}\\
\overline{C}_1 &:= \Theta_2^{-\frac{1}{2}} \Theta_3; \ \overline{C}_2 := H; \ \overline{D}_{21} := \frac{\alpha}{\sqrt{1 - \alpha^2}} I\\
\overline{D}_{22} &:= \frac{\alpha^2}{1 - \alpha^2} I
\end{aligned}
$$

$$
\begin{aligned}
\Theta_4 &:= I - \overline{C}_1\overline{Z}\overline{C}_1^T \\
E_1 &:= \overline{C}_1\overline{Z}\overline{C}_1^T \\
E_2 &:= \overline{C}_2\overline{Z}\overline{A}^T + \overline{D}_{21}\overline{B}_1^T \\
E_3 &:= \overline{C}_1\overline{Z}\overline{C}_2^T \\
\Theta_5 &:= \overline{D}_{21}\overline{D}_{21}^T + \overline{C}_2\overline{Z}\overline{C}_2^T + E_3^T\Theta_4^{-1}E_3 \\
\Theta_6 &:= E_2 + E_3^T\Theta_4^{-1}E_1
\end{aligned}
$$

With above definitions, the following theorem is then suggested for H_∞ ILC.

Theorem 3.1: For the higher-order ILC system given by (8), (9), (10), if there exists $\overline{X} > 0$ and $F = -\Theta_2^{-1}\Theta_3$ such that $\Theta_1 > 0$ and $I + \alpha^2\overline{X}(I - \alpha^2\overline{X})^{-1}\overline{X} - \Theta_3^T\Theta_2^{-1}\Theta_3 < 0$, and also if there exists $\overline{Z} > 0$ such that $\Theta_4 > 0$ and $\overline{A}\overline{Z}\overline{A}^T - \overline{Z} + \overline{B}_1\overline{B}_1^T + E_1^T\Theta_4^{-1}E_1 - \Theta_6^T\Theta_5^{-1}\Theta_6 < 0$, then the observer gain matrix $L := -\Theta_6^T\Theta_5^{-1}$ and the controller given as

$$\xi_{k+1} = A_c\xi_k + B_cY_k$$
$$V_k = C_c\xi_k$$

where $A_c := \overline{A} + (\overline{B}_2 + L\overline{D}_{22})F + L\overline{C}_2$, $B_c = L$, and $C_c = F$, stabilize the system (8), (9), (10) and guarantee $\|T_{ZW}\|_{\infty^w} < 1$.

Proof: The proof is immediate by several algebraic simplifications after substituting

$$A = I;\ B_1 = [\alpha I\ 0];\ B_2 = I;\ C_1 = [H,\ 0]^T;\ C_2 = H;$$
$$D_{11} = \begin{bmatrix} 0 & \alpha I \\ 0 & 0 \end{bmatrix};\ D_{12} = [0,\ I]^T;\ D_{21} = [0,\ \alpha I];\ D_{22} = 0$$

into the system given in Theorem 3.3 of [21]. ■

Remark 3.1: In the controller defined in Theorem 3.1, the dimensions of A_c, B_c, and C_c are the same as that of the Markov matrix H. This implies that the controller $C(w)$ will be of order N in iteration.

Remark 3.2: To solve for the controller matrices A_c, B_c and C_c, in Theorem 3.1, two discrete Riccati type equations, (11) and (12), have to be solved to find the positive definite matrices $\overline{X}$ and $\overline{Z}$. From Remark 3.2 of [21] we can use a recursive solution searching method to solve these Riccati equations. The following equations can be used for this purpose:

$$(\overline{X})_{i+1} = (\overline{X})_i + H^TH + \alpha^2\overline{X}(I - \alpha^2\overline{X})^{-1}\overline{X} + \varepsilon I \quad (11)$$
$$(\overline{Z})_{i+1} = (\overline{Z})_i + \alpha^2 I + E_1\Theta_1^{-1}E_1 - \Theta_6^T\Theta_5^{-1}\Theta_6 + \varepsilon I \quad (12)$$

where ε is fixed as 10^{-5}. These equations iterate starting with $(\overline{X})_0 = (\overline{Z})_0 = 0$, under the condition that all conditions of Theorem 3.1 are satisfied. The final outcome of the iteration is chosen as the solutions of Riccati equations.

B. Model Uncertainty

We now extend the results of the previous section to include the model uncertainty ΔH in system (8), (9), (10). Fig. 4 shows the ILC block diagram with model uncertainty and with the performance penalty given as $Z_k = D_{12}V_k + C_1U_k$. In this case, the input plant sensitivity matrix (from d_I to Z_k) is defined as:

$$
\begin{aligned}
S_I &= C(w)H_p(w)(I + C(w)H_p(w))^{-1} \\
&\quad + (wI - I)^{-1}(I + C(w)H_p(w))^{-1} \qquad (13)
\end{aligned}
$$

and the output plant sensitivity matrix (from d_o to Z_k) is defined as:

$$
\begin{aligned}
S_0 &= C(w)(I + C(w)H_p(w))^{-1} \\
&\quad + C(w)((wI - I)(I + H_p(w)C(w)))^{-1}. \qquad (14)
\end{aligned}
$$

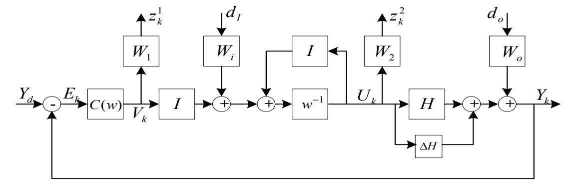

Fig. 4. ILC H_∞ diagram with model uncertainty.

Then, the purpose of H_∞ synthesis is to minimize those sensitivity matrices and to robust stabilize the system with uncertainty ΔH. From Fig. 4, the ILC system can be expressed in the state-space form like:

$$
\begin{aligned}
U_{k+1} &= IU_k + B_1W_k + IV_k && (15) \\
Z_k &= C_1U_k + D_{11}W_k + D_{12}V_k && (16) \\
Y_k &= (H + \Delta H)U_k + D_{21}W_k + D_{22}V_k, && (17)
\end{aligned}
$$

where $Z_k = [z_k^1, z_k^2]^T$, $W_k = [d_I, d_o]^T$, $A = I$, $B_1 = [\alpha I, 0]$, $B_2 = I$, $C_1 = [0,\ I]^T$, $D_{11} = 0_{2n\times 2n}$, $D_{12} = [I,\ 0]^T$, $C_2 = H + \Delta H$, $D_{21} = [0, \alpha I]$, and $D_{22} = 0$. In this case (I, I) is stabilizable, (H, I) is detectable, D_{12} is full column rank, and D_{21} is full row rank. Thus all the basic assumptions for the existence of an H_∞ solution are satisfied.

Remark 3.3: In Fig. 3, if the model uncertainty is included in H, then the system is changed as:

$$
\begin{aligned}
U_{k+1} &= IU_k + B_1W_k + IV_k && (18) \\
Z_k &= (H + \Delta H)U_k + D_{11}W_k + D_{12}V_k && (19) \\
Y_k &= (H + \Delta H)U_k + D_{21}W_k, && (20)
\end{aligned}
$$

So, in this case, the performance penalty Z_k depends on ΔH. But this makes it difficult to standardize the ILC in a typical H_∞ framework. Thus, to overcome this drawback, we used Fig. 4.

To solve the H_∞ robust performance problem with model uncertainty algebraically, an existing result can be used, which is summarized in what follows. From [22], [23], with $D_{22} = 0$, when the following uncertain system is given,

$$
\begin{aligned}
x_{k+1} &= (A + \Delta A)x_k + B_1w_k + (B_2 + \Delta B_2)u_k && (21) \\
z_k &= C_1x_k + D_{11}w_k + D_{12}u_k && (22) \\
y_k &= (C_2 + \Delta C_2)x_k + D_{21}w_k && (23)
\end{aligned}
$$

after making an equivalent auxiliary system (given in equations (24), (25), and (26), below) the robust stability analysis can be performed as done above for the system given by equations (8,9,10). Specifically, let the uncertain system have the following form:

$$\begin{bmatrix} \Delta A & \Delta B_2 \\ \Delta C_2 & \Delta D_{22} \end{bmatrix} = \begin{bmatrix} M_1 \\ M_2 \end{bmatrix} F \begin{bmatrix} N_1 & N_2 \end{bmatrix}$$

where M_1, M_2, N_1, N_2 are constant matrices, and $FF^T \le I$. For a parameter $\epsilon > 0$, then by introducing the auxiliary system,

$$
\begin{aligned}
x_{k+1}^a &= Ax_k^a + H_1w_k^a + B_2u_k^a && (24) \\
z_k^a &= E_1x_k^a + D_{11}w_k^a + E_2u_k^a && (25) \\
y_k^a &= C_2x_k^a + H_2w_k^a + D_{22}u_k^a && (26)
\end{aligned}
$$

where $H_1 = \left[\sqrt{\epsilon}M_1, B_1\right]$, $H_2 = \left[\sqrt{\epsilon}M_2, D_{21}\right]$,

$$E_1 = \begin{bmatrix} \frac{N_1}{\sqrt{\epsilon}} \\ C_1 \end{bmatrix};\ E_2 = \begin{bmatrix} \frac{N_2}{\sqrt{\epsilon}} \\ D_{12} \end{bmatrix}$$

the original uncertain system is stabilizable with $\|G_{zw}\| < 1$ if and only if the auxiliary system is stabilizable with $\|G_{z^a w^a}\|_{\infty^w} < 1$.

Applying this result to our problem, from Fig. 4 define

$$\begin{bmatrix} 0 & 0 \\ \Delta H & 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \Delta H \end{bmatrix} I \begin{bmatrix} I & 0 \end{bmatrix},$$

where $M_1 = 0$, $M_2 = \Delta H$, $F = I$, $N_1 = I$, and $N_2 = 0$. Then typical H_∞ synthesis can be performed with the augmented ILC system given as:

$$x^a_{k+1} = Ax^a_k + B_1 w^a_k + B_2 u^a_k \quad (27)$$
$$z^a_k = C_1 x^a_k + D_{11} w^a_k + D_{12} u^a_k \quad (28)$$
$$y^a_k = C_2 x^a_k + D_{21} w^a_k + D_{22} u^a_k \quad (29)$$

where $A = I$, $B_1 = [0, \alpha I, 0]$, $B_2 = I$, $C_1 = [\frac{I}{\sqrt{\epsilon}}, 0, I]^T$, $D_{11} = 0_{3n \times 3n}$, $D_{12} = [0, I, 0]^T$, $C_2 = H$, $D_{21} = [\sqrt{\epsilon}\Delta H, 0, \alpha I]$, and $D_{22} = 0$. As done in Theorem 3.1 of this paper, Theorem 3.3 of [21] can be modified. This process is quite messy but straightforward.

IV. Simulation Illustrations

Consider the following discrete system:

$$x_{k+1} = \begin{bmatrix} 0.25 & 0.6 \\ 0.6 & 0 \end{bmatrix} x_k + \begin{bmatrix} 1.0 \\ 0.0 \end{bmatrix} u_k$$
$$y_k = [1.0 \quad -1.3] x_k$$

which has nominal eigenvalues at -0.5 and 0.75. In the time axis, 50 discrete samples are used and in the iteration axis, 100 times iteration tests are performed. For this system, after calculating Markov matrix H, the maximum model uncertainty is assumed to be 10 percent of the nominal H (note, for convenience we do not pick up plants with uncertainty $\Delta H(w)$, but rather use plants with interval uncertainty; however, this is acceptable as our plants still satisfy a norm-bound of the form $\|\Delta H\|_{\infty^w} < \epsilon_H$). The external disturbances satisfy $\|\cdot\|_{l_{2^w}} < \alpha = 0.1$ In E_1, E_2, H_1 and H_2 of (24), (25) and (26), ϵ is fixed at 1. Simulation tests were performed for the case of external disturbances alone and for the case of both external disturbances together with model uncertainty. For the latter case we used the augmented system, which is given in (27), (28), (29) with Fig. 4. The controller was designed based on Theorem 3.1. To check the performance of the suggested algebraic method, we also did simulation tests using the controller designed from MATLAB $dhinf$ command, and also using first-order ILC learning gain matrices. For the first-order ILC case, we used both Arimoto-like learning gains (tuned as the best manually) and simply the inverse of the nominal plant (i.e., $C(w) = H^{-1}$). Simulation tests for the case of external disturbances with no model uncertainty are shown in Fig. 5. This figure shows the result of simulating the different ILC controllers with the nominal plant, for fixed l_{2^w} signals $d_I(w)$ and $d_o(w)$. Note, though these signals are iteration-varying, from the perspective of the space l_{2^w} they are fixed over the course of a simulation test. The left plots of Fig. 5 are the ILC performance designed from the MATLAB $dhinf$ and from the inverse of H along the iteration axis. The right plots of Fig. 5 are the results from the suggested algebraic method and from Arimoto-like gains. From these figures, it is not clear to compare the performance differences between the different controller options. Note that the error does not go to zero as the number of iterations increase. That is because the goal is to have minimized the gain between the disturbances and the error. By assumption it cannot be driven to zero (though, the error between Y_d and the output is in fact going to zero). But, the error will depend on the signals d_I and d_o. Thus, to compare the controllers it is useful to consider the actual gain between the disturbances and the error. These are shown in Fig. 6 for a number of different plants defined by our uncertainty model. Recall, the signal norms of interest and the gain is computed over iterations. Thus, for a given simulation (a single plant and a single set of l_{2^w} disturbance signals) we get a single number defining the norm of the error as well as the norm of the disturbance. Fig. 6 plots these norms and their ratio for different plants (chosen randomly from the uncertainty model). Also recall that the algebraic H_∞ controller and MATLAB based H_∞ controller were designed to achieve $\|T_{EW}\|_{\infty^w} < 1$ (but that the Arimoto-like gains and the inverse of H do not guarantee $\|T_{EW}\|_{\infty^w} < 1$). However, this is only guaranteed for the nominal plant. We have not solved the complete, simultaneous robust stability and robust performance problem. In Fig. 6, we plotted $\|E_k\|_{2^w}$ and $\|W_k\|_{2^w}$ as well as their ration for 50 plants. From sub-figures (a) and (b) (ILC systems designed from Arimoto-like gains and $C(w) = H^{-1}$, respectively), we observe that the robust performance requirement $\|T_{EW}\|_{\infty^w} < 1$ is not achieved for many cases. However from (c) and (d) (ILC systems designed from MATLAB and the algebraic method, respectively), we observe that $\|T_{EW}\|_{\infty^w} < 1$ is achieved for most plants (exception due to ΔH, as noted). Clearly, from these figures, we conclude that the ILC system designed based on H_∞ methods are more robust than the first order ILC systems.

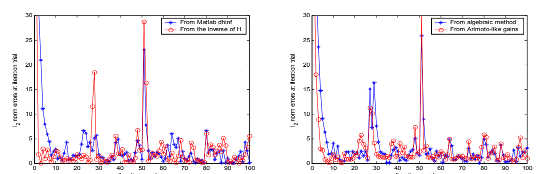

Fig. 5. H_∞ ILC test results.

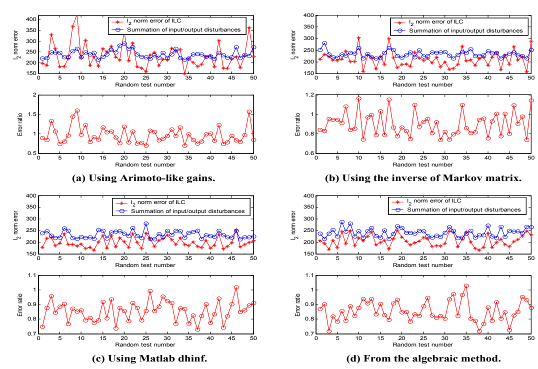

Fig. 6. The summation of the plant input and output disturbances ($(\alpha I)d_I$ and $(\alpha I)d_o$) and $\|(T_{EW})W_k\|_{2^w}$.

V. Conclusions and Further Discussion on H_∞ ILC

In the following we summarize our work and provide our continuing efforts related to this problem:

- In this paper, we provided a new framework to design ILC controllers using H_∞, which can be effectively used for iteration-varying ILC systems.

- In our approach the H_∞ design was done using the discrete frequency domain along iteration axis (not the time axis), in the super-vector framework.

- In this paper, we did not use any filtering to characterize the input and output disturbances. So, there was no way to further reduce the baseline errors. We are currently studying the use of weighting filters that can be used to reduce the baseline error in the iteration domain.

- With structured model uncertainty, μ synthesis can be used to reduce the conservativeness.

- The algebraic H_∞ design is much faster than MATLAB based H_∞ and enables us to understand the H_∞ mechanism in ILC.

The final remark follows. The suggested H_∞ ILC approach can be effectively used to reduce the baseline errors, but in this case W_i and W_o in Fig. 3 and Fig. 4 should be discrete filters. Currently, we are doing this work and the result will be presented in the near future.

REFERENCES

[1] S. Kawamura S. Arimoto and F. Miyazaki, "Bettering operation of robots by learning," *Journal of Robotic Systems*, vol. 1, pp. 123–140, 1984.

[2] K. L. Moore, *Iterative learning control for deterministic systems*, Advances in Industrial Control. Springer-Verlag, 1993.

[3] H. Hashimoto and J.-X. Xu, "Learning control systems with feedback," in *Proceedings of the IEEE Asian Electronics Conference*, Hong Kong, September 1987.

[4] YangQuan Chen and Changyun Wen, *Iterative Learning Control: Convergence, Robustness and Applications*, vol. LNCIS-248 of *Lecture Notes series on Control and Information Science*, Springer-Verlag, London, 1999.

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

[6] D. de Roover, "Synthesis of a robust iterative learning controller using an H_∞ approach," in *Proceedings of the 35th IEEE Conference on Decision and Control*, Kobe, Japan, December 1996.

[7] D. de Roover and Ollo H. Bosgra, "Synthesis of robust multivariable iteraive learning controller with application to a wafer stage motion system," *International Journal of Control*, vol. 73, no. 10, pp. 968–979, 2000.

[8] M. Norrlöf, *Iterative Learing Control: Analysis, Design, and Experiments*, Ph.D. thesis, Linköpings Universitet, Linköping, Sweden, 2000.

[9] S. S. Saab, "A discrete-time stochastic learning contorl algorithm," *IEEE Trans. on Automatic Control*, vol. 46, no. 6, pp. 877–887, 2001.

[10] B. K. Kim, W. K. Chung, and Y. Youm, "Robust learning control for robot manipulators based on disturbance observer," in *IECON Proceedings (Industrial Electronics Conference)*, Taipei, Taiwan, 1996, vol. 2, pp. 1276–1282.

[11] Y. Chen, C. Wen, J.-X. Xu, and M. Sun, "An initial state learning method for iterative learning control of uncertain time-varying systems," in *Proc. of the 35-th IEEE CDC*, Kobe, Japan, Dec. 1996, pp. 3996–4001.

[12] W. Xu, C. Shao, and T Chai, "Learning control for a class of constrained mechanical systems with uncertain disturbances," in *Proceedings of the 1997 American Control Conference*, Albuquerque, NM, June 1997.

[13] Y.-J. Liang and D. P. Looze, "Performance and robustness issues in iterative learning control," in *Proceedings of 32nd IEEE Conference on Decision and Control*, San Antonio, TX, December 1993, pp. 1990–1995.

[14] E. Rogers and D. H. Owens, *Stability Analysis for Linear Repetitive Processes*, Springer-Verlag, Berlin, 1992.

[15] J. B. Edwards and D. H. Owens, *Analysis and Control of Multipass Processes*, Research Studies Press, Taunton, Chichester, 1982.

[16] 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.

[17] 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.

[18] YangQuan Chen and K. L. Moore, "PI-type iterative learning control revisited," in *2002 American Control Conference*, Anchorage, Alaska, USA., May 8-10 2002, AACC, pp. 2138–2143.

[19] YangQuan Chen and Kevin L. Moore, "Harnessing the nonrepetitiveness in iterative learning control," in *Proceedings of 40th IEEE Conference on Decision and Control*, Las Vegas, NE, USA, Dec. 2002, IEEE.

[20] D. J. N. Limebeer, M. Green, and D. Walker, "Discrete time H_∞ control," in *Proceedings of the 28th Conference on Decision and Control*, Tampa, Florida, USA, December 1989, IEEE, pp. 392–396.

[21] K. Furuta and S. Phoojaruenchanachai, "An algebraic approach to discrete-time H_∞ control problems," in *Proc. American Control Conference*, San Diego, CA, USA., May 1990, AACC, pp. 3067–3072.

[22] Carlos E. de Souza, Minyue Fu, and Lihua Xie, "H_∞ analysis and synthesis of discrete-time systems with time varying uncertainty," *IEEE Transactions on Automatic Control*, vol. 38, no. 3, pp. 459–462, 1993.

[23] Lisong Yuan, Luke E. K. Achenie, and Weisun Jiang, "Robust H_∞ control for linear discrete-time systems with norm-bounded time varying uncertainty," *Systems and Control Letters*, vol. 27, no. 4, pp. 199–208, 1996.