Robust Gradient-based Iterative Learning Control

D.H. Owens, J. Hätönen, S. Daley
Department of Automatic Control and Systems Engineering,
University of Sheffield,
Mappin Street, Sheffield S1 3JD, United Kingdom
Email: D.H.Owens@sheffield.ac.uk

January 31, 2007

Abstract

This paper considers the use of matrix models and the robustness of a gradient-based Iterative Learning Control (ILC) algorithm using fixed learning gains to ensure monotonic convergence with respect to the mean square value (Euclidean norm) of the error time series. The paper provides a complete and rigorous analysis for the systematic use of matrix models in ILC. They provide necessary and sufficient conditions for robust monotonic convergence and permit the construction of sufficient frequency domain conditions for robust monotonic convergence on finite time intervals for both causal and non-causal controller dynamics.

Keywords: Iterative learning control, robust control, parameter optimization, positive-real systems

— —

1 Introduction

Iterative Learning Control (abbreviated to ILC in the sequel) is concerned with the performance of systems that operate in a repetitive manner and includes examples such as robot arm manipulators and chemical batch processes, where the task is to follow some specified output trajectory in a specified time interval with high precision. ILC uses information from previous executions of the task in an attempt to improve performance from repetition to repetition in the sense that the tracking error (between the output and the specified reference trajectory) is sequentially reduced to zero (see [1] and [8]). Note that repetitions are often called trials, passes or iterations in the literature.

This paper introduces the idea of gradient-based ILC algorithms for discrete-time systems. These algorithms have the mathematical structure of a nonlinear discrete-time system in a high dimensional space. The paper analyses the behaviour and robustness of these nonlinear algorithms in a rigorous manner. Note that the analysis of continuous-time gradient based algorithms have been carried out in [3] and [7]. In this paper, robustness is defined in terms of a new concept of *Robust Monotone convergence* introduced by the authors in [4]:

Definition: An ILC algorithm has the property of robust monotone convergence with respect to a vector norm $||\cdot||$ *in the presence of a defined set of model uncertainties if, and only if, for every choice of control on the first trial (and hence for every choice of initial error) and for any choice of model uncertainty within the defined set, the resulting sequence of iteration error time signals converges to zero with a strictly monotonically decreasing norm.*

The requirement of monotonicity is representative of a practical requirement to improve tracking from trial to trial. The mean square value of the error time series is used as a norm as it will be seen that it has useful analytical properties in generating checkable design conditions.

A companion paper [4] uses the idea of an inverse model-based algorithm with learning gain $\beta \in (0,1)$ with excellent results if the plant model mismatch is zero but, in the presence of a multiplicative uncertainty (with transfer function $U(z)$), robust monotone convergence is ensured if

$$|\frac{1}{\beta} - U(z)| < \frac{1}{\beta}, \quad \forall\, |z| = 1 \tag{1}$$

A simple analysis of this expression indicates that:

1. significant high frequency errors such as high frequency parasitic resonant modes will require small values of learning gain β and hence slow convergence of the algorithm.

2. In addition, the phase of the uncertainty must lie in the open range $(-\frac{\pi}{2}, \frac{\pi}{2})$, a fact that constrains the form of uncertainty that can be tolerated. It arises from the monotonicity requirement and is equivalent to $U(z)$ being strictly positive real in the sense that $Re[U(z)] > 0, |z| = 1$.

3. If $U(z)$ is not known but is known to belong to the set characterized by an inequality of the form
$$|\frac{1}{\beta^*} - U(z)| < \frac{1}{\beta^*}, \quad \forall\, |z| = 1 \tag{2}$$
then robust monotone convergence is guaranteed for all choice of gains in the range $0 < \beta < \beta^*$ (see [9] for a more extensive review of this topic).

In contrast, for a process with transfer function $G(z) = G_0(z)U(z)$ where $G_0(z)$ is a nominal model used for control purposes, this paper will show that the proposed gradient-based algorithm is robust monotone convergent if
$$|\frac{1}{\beta} - |G_0(z)|^2 U(z)| < \frac{1}{\beta}, \quad \forall\, |z| = 1 \tag{3}$$
This does not remove the need for a strictly positive real $U(z)$. It can however remove the destabilizing effect of high frequency errors as, in practice, both $G(z)$ and $G_0(z)$ are low pass filters and hence $G_0(z)$ will be small at high frequencies.

This paper derives the basic relationships for robust monotone convergence in the case of a constant learning gain β. The choice of optimal gains will be addressed elsewhere.

Following a formal definition of the problem, a "static" matrix model of the dynamic process is introduced. This model (used elsewhere as in [4]) makes analysis simpler than analysis using the state space model directly but, for the purposes of this paper, requires the derivation of a number of algebraic properties of such models. These properties are very useful for manipulation and interpretation purposes in both the time and frequency domain.

The gradient- based algorithm is then introduced firstly in the absence of modelling errors and then in the presence of multiplicative modelling errors. The results are expressed initially in terms of matrix inequalities and then in frequency domain terms using the transfer function description of plant model and uncertainty.

Where appropriate, the paper compares the inverse-model and gradient-based algorithms with the conclusion that the gradient-based approach will be more robust both in theory and in practice.

2 Problem definition

As a starting point consider a standard discrete-time,linear, time-invariant single-input, single-output state-space representation defined over a *finite*, *discrete* time interval, $t \in [0, N]$ (in order to simplify notation it is assumed that the sampling interval, t_s is unity). The system is assumed to be operating in a repetitive mode where at the end of each repetition, the state is reset to a specified *repetition-independent* initial condition for the next operation during which a new control signal can be used. A reference signal $r(t)$ is assumed to be specified and the ultimate control objective is to find an input function $u^*(t)$ so that the resultant output function $y(t)$ tracks this reference signal $r(t)$ *exactly* on $[0, N]$. The process model is written in the form:

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) \qquad x(0) = x_0 \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{4}$$

where t is the sample number, the state $x(\cdot) \in \mathbb{R}^n$, output $y(\cdot) \in \mathbb{R}$ and input $u(\cdot) \in \mathbb{R}$. The operators A, B and C are constant matrices of appropriate dimensions and D is a scalar. From now on it will be assumed that either $D \neq 0$ or that $CA^{j-1}B = 0, 1 \leq j < k^*$ and $CA^{k^*-1}B \neq 0$ for some $k^* \geq 1$ (trivially satisfied in practice) and that the system (4) is both controllable and observable. If $D \neq 0$, then take $k^* = 0$. By construction, k^* is then the relative degree of the transfer function $G(z)$ of the system. Also, the notation $f_k(t)$ will denote the value of a signal f at sample interval t on iteration k.

The repetitive nature of the problem opens up possibilities for modifying iteratively the input function $u(t)$ so that, as the number of repetitions increases, the system asymptotically learns the input function that gives perfect tracking. To be more precise, the control objective is to find a causal recursive control law typified by a relationship of the form

$$u_{k+1}(t) = f(u_k(\cdot), u_{k-1}(\cdot), \ldots u_{k-r}(\cdot), e_{k+1}(\cdot), e_k(\cdot), \ldots, e_{k-s}(\cdot)) \tag{5}$$

with the properties that, independent of the control input time series chosen for the first trial, the resultant sequence of error and input signals satisfy

$$\lim_{k\to\infty} \|e_k(\cdot)\| = 0 \quad \lim_{k\to\infty} \|u_k(\cdot) - u^*(\cdot)\| = 0 \tag{6}$$

where $\| \cdot \|$ denotes any norm for the time series. In what follows, this norm is taken to be the Euclidean norm $||f|| = \sqrt{f^T f}$ in $\mathcal{R}^p$ which is related to the mean square error of the time series by the multiplier $\sqrt{p}$.

3 Matrix Representations of Plant Dynamics

The state space model is a natural description for the *dynamic* process. For this paper, it is argued that an equivalent "static" matrix description is more suited to the method of analysis. More precisely, as the linear system maps input time series into output time series, it follows that there exists a matrix relating these time series. This matrix is an equivalent description of the systems dynamics.

The idea of matrix models is not new (see for example [4]) but their use in analysis has been limited to computation. The contribution of this paper is that it provides a complete and rigorous construction of the formal mathematical structures that are necessary to link matrix models to time and frequency domain methods and system compositions. To construct this matrix model in $\mathbb{R}^{N+1}$, define the time series "super-vectors" on the k^{th} trial via

$$u_k = [u_k(0), u_k(1), \ldots, u_k(N)]^T \tag{7}$$

$$y_k = [y_k(0), y_k(1), \ldots, y_k(N)]^T \tag{8}$$

$$r = [r(0), r(1), \ldots, r(N)]^T \tag{9}$$

$$e_k = [e_k(0), e_k(1), \ldots, e_k(N)]^T = r - y_k \tag{10}$$

Furthermore, let u^* be the input sequence (in time series or supervector form) that gives $r(t) = [G_c u^*](t)$ where G_c is the convolution mapping corresponding to the process model (4).

Note that if the mapping f in (5) is not a function of e_{k+1}, then it is typically said that the algorithm is of *feedforward* type. If it does not depend on any of the $e_j, 0 \leq j \leq k$, it is of feedback type. Otherwise it is of *feedback plus feedforward* type.

With the above definitions, the relevant formulae for the input-output response of the system can be written in the form, $k \geq 0$,

$$y_k = G_e u_k + d_0 \tag{11}$$

where G_e has dimension $(N+1) \times (N+1)$ and the lower triangular band structure $(G_e)_{ij} = (G_e)_{(i+1)(j+1)}$ that is required by causality and time invariance of linear time-invariant convolution systems i.e.

$$G_e = \begin{bmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \dots & \dots & D \end{bmatrix} \tag{12}$$

Also $d_0 = [Cx_0, CAx_0, ..., CA^N x_0]^T$.

The elements CA^jB of the matrix G_e are the Markov parameters of the plant (4). Suppose that the plant transfer function $G(z) = C(zI - A)^{-1}B + D$ has relative degree (pole-zero excess) $k^* \geq 0$. Assume also that the reference signal $r(t)$ satisfies $r(j) = CA^j x_0$ for $0 \leq j < k^*$ (or, alternatively, that tracking in this interval is not important). Then (in a similar manner to [6]) it is noted that, for analysis, it is sufficient to analyse a 'lifted' plant equation that is just the above if $k^* = 0$ or, if $k^* \geq 1$,

$$y_{k,l} = G_{e,l}u_{k,l} + d_1 \tag{13}$$

where the signals u, y, e, r etc are modified to reflect these changes. For example, $u_{k,l} = [u_k(0), u_k(1),\ \dots,\ u_k(N - k^*)]^T$, $y_{k,l} = [y_k(k^*)\ y_k(2)\ \dots\ y_k(N)]^T$ etc and

$$G_{e,l} = \begin{bmatrix} CA^{k^*-1}B & 0 & 0 & \dots & 0 \\ CA^{k^*}B & CA^{k^*-1}B & 0 & \dots & 0 \\ CA^{k^*+1}B & CAk^*B & CA^{k^*-1}B & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \dots & \dots & CA^{k^*-1}B \end{bmatrix} \tag{14}$$

with $d_1 = [CA^{k^*}x_0, ..., CA^N x_0]^T$. For notational convenience, the subscripts e, l are dropped and the model is written in all cases $k^* \geq 0$ in the simplified notational form

$$y_k = Gu_k + d \tag{15}$$

which has the structure of discrete dynamics in $\mathbb{R}^{N+1-k^*}$. Note that:

1. G is invertible by construction which confirms that, for an arbitrary reference r on $0 \leq j \leq N$, there exists a time series u^* on $0 \leq j \leq (N + 1 - k^*)$ such that $r = Gu^* + d$ on $k^* \leq j \leq N$.

2. A comparison of G with G_e indicates that G can be identified with a plant with transfer function $G^*(z) = z^{k^*}G(z)$ operating on an interval $0 \leq j \leq N + 1 - k^*$.

3. An examination of G_e or G indicates that higher order Markov parameters do not appear in the matrix model. As a consequence, the system is indistinguishable from any of the Finite Impulse Response (FIR) models with transfer function

$$G_M(z) = D + \Sigma_{j=1}^{M} CA^{j-1}Bz^{-j}, \quad M \geq N \tag{16}$$

   As a consequence, in what follows, it is always possible to replace transfer functions by FIR equivalents during analysis and/or design.

From now on this lifted plant model will be used as a starting point for analysis and the identification of the matrix G with the transfer function $G^*(z)$ will be used as required.

Let F be the (right-shift) matrix with elements $F_{ij} = \delta_{i,j+1}$

$$F = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix} \tag{17}$$

so that

$$F^j \neq 0, \quad 0 \leq j \leq N - k^* \qquad , \qquad F^j = 0 \quad \forall \quad j \geq N + 1 - k^* \tag{18}$$

A simple calculation then indicates that

$$G = \Sigma_{j=1}^{N+1-k^*} g_j F^{j-1} \tag{19}$$

for suitable choice of scalars $\{g_j\}$. It is also true that all such matrices can be identified (non-uniquely) with linear time invariant systems. Let

$$\mathcal{L}_l = \{G \in \mathcal{R}^{l \times l} : \exists \{g_j\}_{1 \leq j \leq l} \quad s.t. \quad G = \Sigma_{j=1}^{l} g_j F^{j-1}\} \tag{20}$$

Then the following statements are easily proven:

$$\{G_1 \in \mathcal{L}_l \quad \& \quad G_2 \in \mathcal{L}_l\} \implies \{G_1 + G_2 \in \mathcal{L}_l\} \tag{21}$$

$$\{G_1 \in \mathcal{L}_l \quad \& \quad G_2 \in \mathcal{L}_l\} \implies \{G_1 G_2 \in \mathcal{L}_l\} \tag{22}$$

$$\{G_1 \in \mathcal{L}_l \quad \& \quad G_2 \in \mathcal{L}_l\} \implies \{G_1 G_2 = G_2 G_1\} \tag{23}$$

$$\{G \in \mathcal{L}_l \quad \& \quad |G| \neq 0\} \implies \{G^{-1} \in \mathcal{L}_l\} \tag{24}$$

In effect, matrix representations obey all of the normal rules of transfer functions in series and parallel connections (provided that they operate on the same underlying time series).

For the purposes of this paper, $\mathcal{L}_l$ has additional useful structure described using the matrix F_0 defined to be the (time-reversal) matrix with elements $F_{ij} = \delta_{i,N-k^*-j}$ i.e.

$$F_0 = \begin{bmatrix} 0 & \dots & \dots & 0 & 1 \\ 0 & \dots & 0 & 1 & 0 \\ 0 & \dots & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \dots & \dots & 0 \end{bmatrix} \tag{25}$$

If $s \in \mathcal{R}^l$ is the column vector of a time series of length l, then $F_0 s$ is a column vector of the same time series but reversed in time i.e. $(F_0 s)_j = s_{l+1-j}$ for $1 \leq j \leq l$. Note that

$$F_0 = F_0^T \quad , \quad F_0^2 = I \tag{26}$$

and hence, after a little manipulation, it is seen that G and G^T are related by the expression

$$G \in \mathcal{L}_l \implies F_0 G F_0 = G^T \tag{27}$$

The important point is that these definitions enable the interpretation of G^T as a dynamical system or simulation. More precisely it is easily proved that:

$$\{\tilde{y} = G^T \tilde{u}\} \Leftrightarrow \{(F_0 \tilde{y}) = G(F_0 \tilde{u})\} \tag{28}$$

In simulation terms: Suppose that $G \in \mathcal{L}_l$. Then the time series $\tilde{y} = G^T \tilde{u}$ is simply the time reversed response of the linear system G (with zero initial conditions) to the time reversal of $\tilde{u}$.

This result is valuable for this paper which considers the basic algorithm described by the *feed-forward* ILC update rule

$$u_{k+1} = u_k + K e_k, \quad K \in \mathbb{R}^{(N+1-k^*) \times (N+1-k^*)} \tag{29}$$

If feedback is required in the algorithm, it is assumed to have been implemented on the plant and included in $G(z)$ and hence G.

Note: in element by element form, this relation is simply

$$u_{k+1}(t) = u_k(t) + \Sigma_{j=1}^{N+1-k^*} K_{t+1,j} e_k(t+j-1+k^*), \quad 0 \le t \le N-k^* \tag{30}$$

For example, with $K = I$ the update law is just

$$u_{k+1}(t) = u_k(t) + e_k(t+k^*), \quad 0 \le t \le N-k^* \tag{31}$$

The matrix K can, in principle, be arbitrary but, in practice, it is assumed that it will be connected with a dynamical system. As a consequence, it is assumed either that

1. $K \in \mathcal{L}_{N+1-k^*}$ generated from a linear, time invariant system model. Ke can then be computed as the time series generated by the response of the state space model of K from zero initial conditions to the time series e or

2. K is the transpose of the matrix description of a linear time invariant system i.e. $K^T \in \mathcal{L}_{N+1-k^*}$ is derived from a linear time invariant model. Any quantity Ke can hence be computed from a simulation although, in real time, the operation would be anti-causal if it were not for the fact that it is applied to already known signals.

The calculations associated with case two above are simple. The first case covers many situations such as the inverse model approach described in [4]. The second covers the case considered in this paper where the choice of

$$K = \beta_{k+1} G^T \tag{32}$$

will be seen to improve robustness, particularly with respect to high frequency modelling errors.

4 A Gradient-based ILC algorithm

The purpose of this section is to introduce the gradient-based algorithm and to provide necessary and sufficient conditions for monotonic convergence of the mean square error to zero in the presence of a specific multiplicative modelling error. These conditions take the form of matrix inequalities that define constraints both on the learning gain that can be used and on the modelling error that can be tolerated. These conditions will be transformed into more useful frequency domain conditions in the following sections.

Using the notation of the previous sections, consider the matrix model $y_k = Gu_k + d, \quad k \ge 0$, where r is the desired reference time series vector, $e_k = r - y_k$ is the error on the k^{th} trial, and the initial control input time series u_0 has been specified with e_0 as the corresponding error. The resultant error is $e_k = r - d - Gu_k$. A simple analysis of $||e_k||^2 = e_k^T e_k$ indicates that the steepest descent direction for the error is just $G^T e_k$ and hence that the feedforward ILC algorithm

$$u_{k+1} = u_k + \beta G^T e_k \tag{33}$$

may be capable of ensuring a monotonic sequence of Euclidean error norms provided that the learning gain $\beta > 0$ is chosen to be sufficiently small.

Note: $G^T e_k$ can be computed from a state space model of G using simulation methods as discussed in the last section. The matrix representation of the problem therefore is not required for practical implementation.

In the following sections, an analysis is undertaken of the effects of the choice of learning gain β. It generates an estimate of an appropriate range in both the case of zero and non-zero modelling errors. Initially, the analysis is in the from of matrix inequalities. Subsequently these will be converted into easily checked expressions in the frequency domain.

5 The Gradient Algorithm: The Case of No Modelling error

A simple calculation reveals that the gradient-based ILC algorithm evolves from its initial error e_0 as follows

$$e_{k+1} = (I - \beta G G^T) e_k, \quad k \geq 0 \tag{34}$$

Noting that $\beta > 0$ by assumption and that

$$||e_{k+1}||^2 = ||e_k||^2 - \beta 2 e_k^T G G^T e_k + \beta^2 e_k^T G G^T G G^T e_k \tag{35}$$

it follows that, as G is nonsingular by construction,

— —

Theorem: Suppose that $\beta > 0$. A necessary and sufficient condition for the gradient-based ILC algorithm to have the monotonicity and convergence properties

1. $||e_{k+1}|| < ||e_k||, \quad \forall\ k \geq 0 \quad \forall\ e_0 \in \mathbb{R}^{N+1-k^*}$
2. $\lim_{k\to\infty} e_k = 0 \quad \forall\ e_0 \in \mathbb{R}^{N+1-k^*}$

in some range $0 < \beta < \beta'$ is that

$$2I > \beta G^T G > 0 \tag{36}$$

— —

*P*roof: $2I > \beta G^T G$ implies the existence of a number $\epsilon > 0$ such that $\beta G G^T G G^T - 2 G G^T < -\epsilon I$. Monotonicity follows from the discussion preceding the statement of the theorem. To prove convergence to zero, simply note that

$$||e_{k+1}||^2 \leq ||e_k||^2 (1 - \beta\epsilon) \quad \forall\ k \geq 0 \tag{37}$$

This completes the proof as $||e_k||$ goes to zero faster than $(1-\beta\epsilon)^{\frac{k}{2}}$. □

The following corollary is easily proved and provides an estimate of the desired range of the learning gain β:

— —

Corollary: Under the conditions of the theorem above, monotone convergence to zero is achieved if, and only if, $0 < \beta \bar{\sigma}^2(G) < 2$ where $\bar{\sigma}(G)$ is the largest singular value of G.

— —

6 The Gradient Algorithm: Robust Monotone Convergence Conditions

Now let $G(z)$ and $G_0(z)$ be transfer functions of the plant and a nominal model respectively. The relative degree of the model G_0 is denoted k^* and the lifted representations (and associated input and output supervectors) are based on this parameter. To ensure that the matrix representations of plant, nominal model and multiplicative perturbations are causal, it is assumed that the relative degree of the plant is equal to or exceeds that of the nominal model.

If there is mismatch between the plant and model, then the gradient-based ILC algorithm is naturally replaced by the approximation

$$u_{k+1} = u_k + \beta G_0^T e_k \tag{38}$$

where G_0 is the lifted matrix representation of a model of $G_0(z)$. The error evolution equation becomes

$$e_{k+1} = (I - \beta G G_0^T) e_k \tag{39}$$

Suppose now that plant and model are related by the expression

$$G(z) = G_0(z) U(z) \tag{40}$$

and $U(z)$ is assumed to be proper and stable. It follows that, if $U(z)$ has a matrix representation U_e (without lifting), then

$$G = G_0 U_e = U_e G_0 \tag{41}$$

Note that $\beta > 0$ by assumption and that

$$\begin{aligned} ||e_{k+1}||^2 &= ||e_k||^2 - \beta e_k^T (G_0 U_e G_0^T + G_0 U_e^T G_0^T) e_k + \beta^2 e_k^T G_0 G_0^T U_e^T U_e G_0 G_0^T e_k \\ &= ||e_k||^2 - \beta e_k^T G_0 [U_e + U_e^T - \beta G_0^T U_e^T U_e G_0] G_0^T e_k \end{aligned} \tag{42}$$

It follows that:

— —

Theorem (Robust Monotone Convergence): The gradient-based ILC algorithm is robust monotone convergent in the presence of the multiplicative modelling error $U(z)$ if, and only if,

$$U_e + U_e^T > \beta G_0^T U_e^T U_e G_0 > 0 \tag{43}$$

— —

Proof: Monotonicity follows trivially from the above noting that G_o is nonsingular by construction. The proof of convergence to zero error follows in a similar way to the previous case. □

— —

Corollary: A necessary condition for monotone robust convergence is that the modelling error matrix representation U_e is positive definite in the sense that $U_e + U_e^T$ is positive definite.

— —

Proof: The proof follows trivially from the observation $\beta G_0^T U_e^T U_e G_0 > 0$. □

Note: The case of no modelling error is retrieved by choosing $U = I$ in the above.

In the next section, more useful frequency domain conditions are provided to check the matrix inequalities derived above.

7 Robustness: Frequency Domain Conditions

In this section the matrix inequalities of the previous sections are converted into sufficient conditions for robust monotone convergence in terms of the transfer functions of the system, model and uncertainty. The practical benefit is that the frequency domain conditions are more easily checked and throw more light on to the benefits and issues facing the application of the gradient-based algorithm.

The approach taken is based on the analysis of matrix inequalities in $\mathbb{R}^{l \times l}$ of the form

$$H_1^T H_1 < H_2 + H_2^T \tag{44}$$

where both $H_1 \in \mathcal{L}_l$ and $H_2 \in \mathcal{L}_l$ are matrix representations of single-input/single-output linear time-invariant systems $H_1(z)$ and $H_2(z)$ on the resultant interval $0 \le j \le l-1$.

The development of frequency domain conditions is based on the idea of examining dynamics on the infinite half interval $[0, \infty)$. Complex integration, positivity and causality then provide the necessary connections.

Let $e = [e(0), e(1), \ldots, e(l-1)]^T$ be a time series of length l and interpret $H_1 e$ as the restriction (to $0 \le j \le l-1$) of the response of $H_1(z)$ (on $[0, \infty)$) to the input with $\mathcal{Z}$-transform $e(z) = \sum_{j=0}^{l-1} e(j) z^{-j}$ i.e. to an infinite sequence $\tilde{e}$ consisting of the l elements of e followed by zeros. Using the fact that the mean square error on a finite interval is always less than or equal to that on the infinite interval, Parseval's Theorem then gives

$$e^T H_1^T H_1 e = ||H_1 e||^2 \le \frac{1}{2\pi i} \oint_{unitcircle} |H_1(z)|^2 |e(z)|^2 \frac{dz}{z} \tag{45}$$

A simple calculation then indicates that

$$||H_1^{-1}||_\infty^{-1} \le \underline{\sigma}(H_1) \le \overline{\sigma}(H_1) \le ||H_1||_\infty \tag{46}$$

where $\underline{\sigma}(H)$ and $\overline{\sigma}(H)$ denote the smallest and largest singular values of a matrix $H \in \mathcal{L}_l$ respectively and $||H||_\infty$ denotes the H_∞ norm of the associated transfer function $H(z)$ on the region $|z| \geq 1$.

In a similar manner, $e^T H_2 e$ is the inner product in l_2 (the space of square summable infinite sequences) of $\tilde{e}$ with the response of $H_2(z)$ to $\tilde{e}$ and hence the exact expression follows from elementary complex variable theory

$$e^T(H_2^T + H_2)e = \frac{1}{2\pi i}\oint_{unitcircle}[H_2(z) + H_2(z^{-1})]|e(z)|^2\frac{dz}{z} \tag{47}$$

The matrix inequality describing robust monotone convergence hence is satisfied if, for all choices of e,

$$\frac{1}{2\pi i}\oint_{unitcircle}|H_1^*(z)|^2|e(z)|^2\frac{dz}{z} \leq \frac{1}{2\pi i}\oint_{unitcircle}[H_2(z) + H_2(z^{-1}]|e(z)|^2\frac{dz}{z} \tag{48}$$

It is now possible to state the following theorem:

— —

Theorem(Robust Monotone Convergence): The gradient-based ILC algorithm using the nominal model $G_0(z)$ is robust monotone convergent in the presence of the multiplicative modelling error with transfer function $U(z)$ if (a sufficient condition)

$$|\frac{1}{\beta} - |G_0(z)|^2U(z)| < \frac{1}{\beta} \quad \forall z \in \{z : |z| = 1\} \tag{49}$$

— —

Proof: The discussion preceding this result and the matrix inequality condition of the previous section indicates that a sufficient condition for robust monotone convergence is that

$$U(z) + U(z^{-1}) > \beta|G_0^*(z)U(z)|^2 \quad \forall |z| = 1 \tag{50}$$

Noting that G_0^* can be replaced by G_0 on $|z| = 1$, multiplying by $\beta|G_0(z)|^2$ and rearranging yields the required result. □

Note: Simple calculations indicate that the frequency domain conditions have a simple and easily checked graphical interpretation, namely that:

The plot of the frequency response function $|G_0(z)|^2U(z)$ on the unit circle $|z| = 1$ lies in the interior of the circle of centre $\frac{1}{\beta}$ and radius $\frac{1}{\beta}$

Recent work by the authors [4] using the inverse model algorithm produced the condition:

$$|\frac{1}{\beta} - U(z)| < \frac{1}{\beta} \quad \forall z \in \{z : |z| = 1\} \tag{51}$$

At its simplest level, the difference between the two results is the replacement of U by $|G_0|^2U$. With this in mind, the use of the gradient-based algorithm can be seen to have the following properties as compared with the inverse-model algorithm:

1. Both approaches require a strictly positive real $U(z)$ for monotone robust convergence. This condition is connected very closely with the monotonicity property of the mean square error and it is expected, as with the inverse-model-based approach, that violation may lead to lack of convergence/instability. Another possibility is that *asymptotic* convergence may be retained but it may also be associated with error norm sequences that can increase from trial to trial.

2. In both cases, the positive real requirement on $U(z)$ will tend to require that it is proper but not strictly proper i.e. that G and G_0 have the same relative degree.

3. The gradient-based algorithm will however reduce performance limitations due to the effect of high frequency errors such as high frequency resonances in G not modelled in G_0. In such circumstances $U(z)$ will tend to take large gain values at frequencies close to these resonances.

This will then require the use of small values of learning gain β to satisfy the monotone convergence criterion for the inverse model algorithm. This does not occur for the gradient-based algorithm because, in practice, G is typically a low pass filter and hence both $G(z)$ and G_0 will be small at high frequencies. The magnitude of $|G_0|^2 U$ will then be substantially reduced (as compared with U) and permit increased learning gains leading to improved convergence rates.

4. In contrast with the beneficial high frequency effects of the gradient-based algorithm, it is possible that it could reduce performance if G (and hence G_0) has a substantial resonance peak within its bandwidth. A similar argument to the above suggests that the learning gains permitted will be reduced (as compared with the inverse model algorithm). As a consequence, it is desirable for a feedback control to be incorporated into the plant (and hence G) before the ILC analysis is undertaken. The feedback controller could be designed along classical lines and, in particular, designed to remove or reduce the resonance peak. In such circumstances, the high frequency benefits of the gradient-based approach indicate that it will, in practice, often be superior to the inverse-model algorithm in terms of its performance and robustness.

5. The above analysis has considered a specific uncertainty U. It can easily be extended to cover sets of multiplicative uncertainties such as any subset of all proper multiplicative uncertainties satisfying an inequality of the form

$$|\frac{1}{\beta^*} - |G_0(z)|^2 U(z)| < \frac{1}{\beta^*} \quad \forall z \in \{z : |z| = 1\} \tag{52}$$

   for some choice of parameter β^*. Clearly robust monotone convergence is achieved in the presence of any model error in this set if $\beta \in (0, \beta^*)$.

In conclusion, the analysis of monotone convergence has been seen to have elegant solutions in terms of inequalities between matrix representations of the plant and associated models. These inequalities can be converted into simple frequency domain (sufficient) conditions that indicate that the gradient-based approach has real potential for both performance and robustness.

Finally, note that, when $U(z) \equiv 1$ and hence $U_e = I$, the above results produce conditions for monotone convergence when there is no plant-model mismatch.

— —

Corollary: Under the conditions of the theorem above, monotone convergence to zero is achieved in the absence of modelling errors if $0 < \beta ||G||_\infty^2 < 2$ where $||G||_\infty = \sup_{|z|=1} |G(z)|$ is the familiar H_∞ norm of G on $\{z : |z| \geq 1\}$.

— —

Proof: Setting $U = I, U(z) \equiv 1$ and $G_0(z) \equiv G(z)$ in the previous result, monotone convergence follows if $|\frac{1}{\beta} - |G(z)|^2| < \frac{1}{\beta} \quad \forall z \in \{z : |z| = 1\}$. The result follows from simple complex algebra. □

In particular, the result shows that, in the absence of mismatch, monotone convergence is not dependent on the phase characteristics of the plant (an observation that links these results to the continous-time methodology described in [10]).

8 Conclusions

The paper has provided a complete analysis of the robust monotone convergence of a gradient-based Iterative Learning Control algorithm in terms of necessary and sufficient matrix inequalities and frequency domain conditions that can be easily checked in terms of plant model and modelling error transfer functions. The method of analysis was the use of matrix models relating the time series of input, output and error signals. A complete analysis of these models is provided which demonstrates that the relative degree of the plant and model are crucial parameters in the analysis of ILC dynamics and hence, it is argued, in the construction of feedforward learning laws. In addition, they clearly show that the use of the "non-causal" gradient operator can be implemented using a plant model and time reversal operations i.e. state space models rather than the matrix models used in the analysis are all that is required for implementation purposes.

The work parallels that published by the authors in a recent paper [4] on inverse-model-based ILC. A comparison with those results indicates that, whereas both approaches require that the multiplicative modelling error has positivity properties (a consequence of the requirement for monotonicity of the mean square error), the gradient approach offers considerable benefits for robustness, particularly in the presence of high frequency modelling errors such as parasitic structural resonance(s).

References

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

[2] S. Daley, J. Hätönen, and D.H. Owens. Hydraulic servo system command shaping using iterative learning control. In *Proceedings of the UKACC Conference, Control 2004*, Bath, UK, 2004.

[3] K. Furuta and M. Yamakita. The design of learning control systems for multivariable systems. In *Proceedings of the IEEE International Symposium on Intelligent Control*, pages 371–376, Philadelphia, Pennsylvania, U.S.A., 1987.

[4] T.J. Harte, J. Hätönen, and D.H. Owens. Discrete-time inverse model-based iterative learning control: stability, monotonicity and robustness. *International Journal of Control*, 78(8):577–586, 2005.

[5] J. Hätönen, T.J. Harte, D.H. Owens, J. Ratcliffe, P. Lewin, and E. Rogers. A new robust iterative learning control law for application on a gantry robot. In *Proceedings of the 9th IEEE conference on Emerging Technologies and Factory Automation*, Lisbon, Portugal, 2003.

[6] J. Hätönen, D.H. Owens, and K.L. Moore. An algebraic approach to iterative learning control. *International Journal of Control*, 77(1):45–54, 2004.

[7] K. Kinosita, T. Sogo, and N. Adachi. Iterative learning control using adjoint systems and stable inversion. *Asian Journal of Control*, 4(1):60–67, 2002.

[8] K.L. Moore. *Iterative Learning Control for Deterministic Systems*. Springer-Verlag, 1993.

[9] D.H. Owens and J. Hätönen. Iterative learning control - an optimization paradigm. *Annual Reviews in Control*, 29(1):57–70, 2005.

[10] Y. Ye and D. Wang. Zero phase learning control using reversed time input runs. *Journal of Dynamic Systems, Measurement and Control*, 127:133–139, 2005.