Convergence and Robustness of a Point-to-Point Iterative Learning Control Algorithm

Thanh V Dinh, Chris T Freeman, Paul L Lewin and Ying Tan

Abstract— **Iterative learning control (ILC) is a methodology applied to systems which repeatedly perform a tracking task defined over a fixed, finite time duration. In this approach the output is specified at all points in this interval, however there exists a broad class of applications in which the output is only important at a subset of time instants. An ILC update law is therefore derived which enables tracking at any subset of time points, with performance shown to increase as time points are removed from the tracking objective. Experimental results using a multi-variable test facility confirm that point-to-point ILC leads to superior performance than can be obtained using standard ILC and an *a priori* specified reference.**

I. INTRODUCTION

For more than 25 years Iterative Learning Control (ILC) has attracted significant interest as an approach applicable to systems which repeatedly track a reference, $\mathbf{y}_d(t)$, defined over a finite interval $0 \leq t \leq T$. Resetting occurs between each repetition, termed a trial, and past experience is employed to improve the tracking accuracy over the subsequent trial. Research has focused jointly on theoretical and application developments, see [1] for a recent review of the literature. Recently ILC has been applied to the 'point-to-point' motion control problem in which an output reference is only prescribed at a finite set of time instants. This form of objective is required in many practical applications, including satellite positioning, rehabilitation engineering, production line automation, and robotic 'pick and place' tasks.

The point-to-point problem can be addressed by the standard ILC framework by simply specifying any reference, $\mathbf{y}_d(t)$, which passes through the critical output points. Improved results, however, are gained if this is coupled with approaches such as Input Shaping which suppress vibrations occurring between the desired points [2]. Alternatively, a simpler feedback controller can be implemented to track $\mathbf{y}_d(t)$ and ILC used to update parameters within the input shaping filter applied to the reference [3]. Another approach is to design an ILC algorithm which not only ensures tracking of $\mathbf{y}_d(t)$, but also attenuates residual vibrations occurring after the point-to-point location is reached [4].

The drawback to all these methods is that an *a priori* designed $\mathbf{y}_d(t)$ is not robust to model uncertainty and noise, and furthermore, they do not capitalize on the freedom available to satisfy additional performance demands.

Point-to-point motion control has also been addressed by approaches that do not employ a static reference defined over $0 \leq t \leq T$, but these have only considered the case where a specified position must be reached at time $t = T$ [5], [6], [7], [8], or the case of a movement between two equilibrium points [9]. Whilst these dispense with tracking unnecessary output points, they do not use the resulting freedom to tackle practical performance objectives, and only consider a single point-to-point movement, rather than a sequence of actions needed to construct complex movements.

This paper addresses current drawbacks by providing a framework that can deal with an arbitrary number of point-to-point movements, while also addressing a performance objective of wide practical utility. The action of embedding both the performance and tracking objective within the framework of iterative learning yields an algorithm which is capable of reaching optimal solutions in the presence of model uncertainty and noise. To achieve this, ILC is employed as an iterative optimization paradigm which uses experimental data to tackle a cost function. A similar approach is applied in [10], but without explicit reference to, or analysis of, the point-to-point tracking problem.

II. PROBLEM FORMULATION

The $n \times n$ identity and zero matrices are denoted by I_n and 0_n respectively. For vector $\mathbf{x} \in \mathbb{R}^n$, $\|\mathbf{x}\|_2 = \sqrt{\mathbf{x}^T\mathbf{x}}$. For matrix $A \in \mathbb{R}^{n\times n}$, $\|A\|$ is the induced norm of the vector norm, $\lambda_i(A)$ denotes the i^{th} eigenvalue of A, $\rho(A) = \max_i |\lambda_i(A)|$ is the spectral radius, $\overline{\sigma}(A)$ and $\underline{\sigma}(A)$ denote the minimum and maximum singular values respectively.

Consider the following linear time-varying (LTV) system

$$\begin{array}{rcl} \mathbf{x}(t+1) & = & A(t)\mathbf{x}(t) + B(t)\mathbf{u}(t) \\ \mathbf{y}(t) & = & C(t)\mathbf{x}(t), \qquad \mathbf{x}(0) = \mathbf{x}_0 \end{array} \tag{1}$$

defined over the finite time interval $t \in [0, 1, 2, \ldots, N-1]$. Here $\mathbf{x}(\cdot) \in \mathbb{R}^n$, $\mathbf{u}(\cdot) \in \mathbb{R}^m$, $\mathbf{y}(\cdot) \in \mathbb{R}^p$ are the state, input and output vectors respectively, and the input and output sequences are given by

$$\begin{array}{rcl} \mathbf{u} & = & [\mathbf{u}(0)^T, \mathbf{u}(1)^T, \ldots, \mathbf{u}(N-1)^T]^T \quad \in \mathbb{R}^{mN} \\ \mathbf{y} & = & [\mathbf{y}(1)^T, \mathbf{y}(2)^T, \ldots, \mathbf{y}(N)^T]^T \quad \in \mathbb{R}^{pN}. \end{array}$$

The standard ILC framework constructs a series of inputs which drives the system to track a reference sequence

$$\mathbf{y}_d = [\mathbf{y}_d(1)^T, \mathbf{y}_d(2)^T, \ldots, \mathbf{y}_d(N)^T]^T \quad \in \mathbb{R}^{pN}.$$

Let $\mathbf{u}_k$ and $\mathbf{y}_k$ be the input and output vectors respectively on the k^{th} trial. Then it is necessary to find a sequence of

T. Dinh, C. Freeman and P. Lewin are with the School of Electronics and Computer Science, University of Southampton, Southampton, SO17 1BJ, UK tdv205@ecs.soton.ac.uk. Y. Tan is with the Electrical and Electronic Engineering Department, University of Melbourne, Parkville, VIC 3010, Australia.

Preprint submitted to 51st IEEE Conference on Decision and Control .
Received March 7, 2012.

control inputs satisfying

$$\lim_{k\to\infty} \|\mathbf{e}_k\| = 0, \qquad \lim_{k\to\infty} \|\mathbf{u}_k - \mathbf{u}_d\| = 0 \tag{2}$$

where $\mathbf{e}_k = \mathbf{y}_d - \mathbf{y}_k$ is the tracking error, and $\mathbf{u}_d$ is the unknown desired input sequence corresponding to $\mathbf{y}_d$. Over the k^{th} trial the relationship between the input and output time-series can be expressed by $\mathbf{y}_k = G\mathbf{u}_k + \mathbf{y}_0$ where $\mathbf{y}_0$ is the response to initial conditions, and $G =$

$$\begin{bmatrix} C(1)B(0) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ C(N)\prod_{i=1}^{N-1} A(i)B(0) & \cdots & C(N)B(N-1) \end{bmatrix} \in \mathbb{R}^{pN\times mN} \tag{3}$$

The effect of $\mathbf{y}_0$ can be absorbed into the reference trajectory, so that it is assumed $\mathbf{y}_0 = \mathbf{0}$, or equivalently $\mathbf{x}_0 = \mathbf{0}$.

A commonly applied ILC update of the form

$$\mathbf{u}_{k+1} = \mathbf{u}_k + L\mathbf{e}_k \tag{4}$$

can be considered as an iterative numerical method to solve the tracking problem, and has been the focus of significant research. It is convergent to a solution if and only if

$$\rho(I_{pN} - GL) < 1. \tag{5}$$

The convergence speed is determined by the magnitude of $\rho(\cdot)$ and is maximum when $\rho(\cdot) = 0$.

A. Point-to-point ILC formulation

Now suppose that the j^{th} plant output is only required to track a reference trajectory at a fixed number, $M_j \leq N$, of sample instants along the trial duration. These sample instants are given by $0 \leq n_{j,1} < n_{j,2} < \cdots < n_{j,M_j} < N$. To define the point-to-point tracking problem it is first necessary to remove the points that do not need to be tracked from the original reference $\mathbf{y}_d$. This yields a reduced reference vector $\mathbf{y}_r \in \mathbb{R}^M$ whose length is given by $M = \sum_{j=1}^{p} M_j$. It is then necessary to define a matrix transformation $\Phi \in \mathbb{R}^{M\times pN}$ such that $\mathbf{y}_r = \Phi\mathbf{y}_d$. This is achieved by first introducing a row vector $\boldsymbol{\psi} \in \mathbb{R}^{pN}$ whose i^{th} element is 1 if the i^{th} element of $\mathbf{y}_d$ is required to be tracked, and 0 otherwise. The formal definition for $\boldsymbol{\psi}$ is

$$\boldsymbol{\psi}_i = \begin{cases} 1 & \text{if} \quad \lfloor (i-1)/p \rfloor \in \mathcal{S}_{i-\lfloor (i-1)/p\rfloor p}, \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\mathcal{S}_j = \{n_{j,1}, \ldots, n_{j,M_j}\}$ and $\lfloor\cdot\rfloor$ denotes the 'floor' function. The matrix Φ is then produced as follows: 1) set $\Phi = \boldsymbol{\psi}$, 2) starting at the first element, increment along the bottom row of Φ, and whenever a non-zero element is encountered move all subsequent bottom row entries into a newly created bottom row that is appended to Φ, maintaining their position along the row and padding the remaining entries of both rows with zeros. Formally this is defined by

$$\Phi_{i,j} = \begin{cases} 1 & \text{if} \quad \boldsymbol{\psi}_j = 1, \sum_{q=1}^{j} \boldsymbol{\psi}_q = i \\ 0 & \text{otherwise} \end{cases}. \tag{7}$$

As seen by the relation $\mathbf{y}_r = \Phi\mathbf{y}_d$, when any output vector is pre-multiplied by Φ, it extracts the components that correspond to prescribed point-to-point locations, while retaining the order in which they appear.

Remark 1: If each output is stipulated at the same set of point-to-point locations, that is $\mathcal{S}_j = \mathcal{S}_1, \ \forall\, j \in \{2,\ldots,p\}$, then matrix Φ has block-wise components

$$\Phi_{i,j} = \begin{cases} I_p & \text{if} \quad j = n_{1,i}, \quad i = 1,2,\ldots M_1 \\ 0_p & \text{otherwise.} \end{cases} \tag{8}$$

In addition the reference has the form

$$\mathbf{y}_r = \left[\mathbf{y}_r(1)^T, \mathbf{y}_r(2)^T, \cdots, \mathbf{y}_r(M_1)^T\right]^T \qquad \in \mathbb{R}^M \tag{9}$$

where $\mathbf{y}_r(i) \in \mathbb{R}^p$ is the prescribed output vector at sample $n_{1,i}$, and $M = pM_1$. □

ILC can be formulated for the point-to-point case by deriving an iterative numerical solution to the problem of finding a control input which minimizes the point-to-point error norm. The control objective is to find a sequence of control inputs $\{\mathbf{u}_k\}$ such that

$$\lim_{k\to\infty} \|\mathbf{y}_r - \Phi G\mathbf{u}_k\| = 0 \tag{10}$$

which replaces the standard requirement (2). The ILC update (4) now assumes the form

$$\mathbf{u}_{k+1} = \mathbf{u}_k + L(\mathbf{y}_r - \Phi\mathbf{y}_k) \tag{11}$$

so that the convergence condition (5) becomes

$$\rho(I_M - \Phi GL) < 1 \tag{12}$$

In Section III a learning operator L is derived to satisfy (12), but first further motivation is provided to support the utility of point-to-point ILC over the standard framework.

B. Point-to-point ILC motivation

Theorem 1: Let d denote the rank deficiency of the plant matrix G (the number of linearly dependent rows). If $d > 0$ the standard ILC update (4) cannot force the plant to track an arbitrary reference trajectory $\mathbf{y}_d$. However the point-to-point update (11) can enforce tracking of an arbitrary reference $\mathbf{y}_r$ if and only if the tracked points are chosen such that

$$M \leq Np - \max\{d, N(p-m)\} \tag{13}$$

Proof. A necessary and sufficient condition for an operator L to exist satisfying (12) is that $\operatorname{rank}(\Phi G) = M$. For the standard ILC case $\Phi = I$, $M = N$ and hence $\operatorname{rank}(\Phi G) = N - d < M$, leading to $I_M - \Phi GL$ having d eigenvalues at unity. Now the i^{th} row of ΦG is the $(j|\Phi_{i,j} = 1)^{th}$ row of G, hence if $p \leq m$ and the point-to-point samples are chosen to correspond to any subset of linearly independent rows of G, the convergence condition (12) can be satisfied. If $p > m$ then the additional condition $M \leq Nm$ is imposed. □

Remark 2: Suppose system (1) is linear time-invariant, and written as the discrete transfer-function matrix $G(z) = C(zI_n - A)^{-1}B + D$ with component $G_{i,j}(z)$ corresponding to the i^{th} output and j^{th} input. If the relative degree of $G_{i,j}(z)$ is $1 + r_{i,j}$, then $d = \sum_{i,j} r_{i,j}$.

The ability of point-to-point ILC to employ a modified standard reference to recover feasibility is extremely important, however many tasks are naturally defined only at a small number of points, and hence additional benefits may also be expected by not enforcing unnecessary tracking. The next lemma shows how the space of feasible inputs expands as the number of tracked points, M, reduces.

Lemma 1: Assuming (13) is satisfied, the feasible input space which forces the system (1) to track $\mathbf{y}_r$ is of dimension $pN - M$, and is given by $\mathcal{U} = \left\{ (\Phi G)^{\dagger} \mathbf{y}_r + \mathbf{x}, \mathbf{x} \in \text{null}\,(\Phi G) \right\}$ where $A^{\dagger} = (A^T A)^{-1} A^T$ is the pseudoinverse of A. The nullspace of ΦG has an orthogonal basis given by the rows of $\overline{\Phi G} P^{\perp}_{(\Phi G)^T}$, where $\overline{\Phi G} \in \mathbb{R}^{(pN-M)\times mN}$ is such that the matrix $\left[\overline{\Phi G}^T (\Phi G)^T \right]$ is full rank, and $P^{\perp}_{A} = I - AA^{\dagger}$ is the orthogonal projection onto the nullspace of A. □

The algorithm proposed in Section III exploits this enlarged feasible input space to achieve desirable performance.

III. GRADIENT DESCENT POINT-TO-POINT ILC

The gradient descent method is one of many numerical algorithms used to tackle nonlinear optimization problems, and has previously been applied within the standard ILC framework [11]. Motivated by (10) and the accompanying discussion, it is applied to solve

$$\min_{\mathbf{u}} J(\mathbf{u}), \qquad J(\mathbf{u}) = \|\mathbf{y}_r - \Phi G \mathbf{u}\|^2 \tag{14}$$

leading to the iterative update for the control input

$$\begin{aligned} \mathbf{u}_{k+1} &= \mathbf{u}_k - \frac{\beta}{2} \nabla_{\mathbf{u}} J(\mathbf{u}_k) \\ &= \mathbf{u}_k + \beta \left(\Phi G\right)^T \left(\mathbf{y}_r - \Phi \mathbf{y}_k\right) \end{aligned} \tag{15}$$

where the experimental plant output, $\mathbf{y}_k$ has replaced the nominal value, $G\mathbf{u}_k$, so that the optimisation is robustly achieved within the ILC framework.

Theorem 2: Let Φ comprise point-to-point locations satisfying $\text{rank}\,(\Phi G) = M$. Let $\tilde{\Phi}$ equal Φ but with the i^{th} row removed, and hence correspond to tracking all but the i^{th} point-to-point location. Let the M eigenvalues of the matrix $A = (\Phi G)(\Phi G)^T$ be denoted $\lambda_M \leq \lambda_{M-1} \cdots \leq \lambda_2 \leq \lambda_1$, which also equal the singular values since A is Normal. Similarly, let the $M-1$ eigenvalues of the matrix $B = (\tilde{\Phi} G)(\tilde{\Phi} G)^T$ be denoted $\mu_M \leq \mu_{M-1} \cdots \leq \mu_3 \leq \mu_2$, which also equal the singular values since B is Normal. Then the following relationship holds

$$\lambda_M \leq \mu_M \leq \lambda_{M-1} \leq \mu_{M-1} \cdots \leq \mu_3 \leq \lambda_2 \leq \mu_2 \leq \lambda_1. \tag{16}$$

In particular, let $\mathbf{y}$ equal the i^{th} column of A with the i^{th} element removed. Then if the eigenvalues of B are distinct and no eigenvector of B is orthogonal to $\mathbf{y}$ then

$$\lambda_M < \mu_M < \lambda_{M-1} < \mu_{M-1} \cdots < \mu_3 < \lambda_2 < \mu_2 < \lambda_1. \tag{17}$$

Proof. First note that A is a Hermitian matrix of order M, and that B is a principal submatrix of A of order $M-1$. Then (16) follows as an application of Cauchy's Interlace Theorem for eigenvalues of Hermitian matrices [12]. It is further proven in [12] that (17) holds provided: 1) the eigenvalues of B satisfy $\mu_M < \mu_{M-1} \cdots < \mu_3 < \mu_2$, and 2) the vector

$$U^T \mathbf{y} = [z_2, z_3, \ldots, z_M]^T \tag{18}$$

has non-zero elements, where U is a unitary matrix of order $M-1$ such that $U^T B U = D$, with $D = \text{diag}\{\mu_2, \mu_3 \ldots \mu_M\}$. To satisfy 2) a suitable choice for U has columns that are the eigenvectors of B, and hence $\exists\, z_i = 0$ only if $\mathbf{y}$ is orthogonal to an eigenvector of B. □

A. Practical Implementation

Remark 3: The term $\left(\Phi G\right)^T \left(\mathbf{y}_r - \Phi \mathbf{y}_k\right)$ in (15) can be efficiently generated using the co-state representation of system (1), and is equal to the output $\tilde{\mathbf{y}}$ of the system

$$\begin{aligned} \tilde{\mathbf{x}}(t) &= A^T(t+1)\tilde{\mathbf{x}}(t+1) + C^T(t+1)\tilde{\mathbf{u}}(t+1) \\ \tilde{\mathbf{y}}(t) &= B^T(t)\tilde{\mathbf{x}}(t) + D^T(t)\tilde{\mathbf{u}}(t), \qquad t = N-1, \ldots, 0 \end{aligned}$$

with the input $\tilde{\mathbf{u}} = \Phi^T \left(\mathbf{y}_r - \Phi \mathbf{y}_k\right)$. This therefore avoids calculation of the large G matrix appearing in (15).

B. Convergence Speed

Theorem 3: Provided the point-to-point locations are chosen to satisfy (13), the choice of gain in (15)

$$0 < \beta < \frac{2}{\overline{\sigma}(\Phi G (\Phi G)^T)} = \frac{2}{\|\Phi G (\Phi G)^T\|} \tag{19}$$

guarantees convergence of the plant output to $\mathbf{y}_r$. In particular, the worse case convergence rate is maximised using

$$\beta = \frac{2}{\overline{\sigma}(\Phi G (\Phi G)^T) + \underline{\sigma}(\Phi G (\Phi G)^T)} \tag{20}$$

The convergence rate using (20) increases as the number of point locations, M, is reduced.

Proof. The convergence condition for (15) corresponds to

$$\rho(I_M - \beta \Phi G (\Phi G)^T) < 1 \tag{21}$$

providing a linear convergence rate to zero error [11]. Since

$$\begin{aligned} \sigma_i(I_M - \beta \Phi G (\Phi G)^T) = 1 - \beta \sigma_i(\Phi G (\Phi G)^T) < 1 \qquad & \forall i \\ v \Rightarrow 0 < \beta \sigma_i(\Phi G (\Phi G)^T) < 2 \qquad & \forall i \end{aligned}$$

(19) follows as $\sigma_i(\Phi G (\Phi G)^T) > 0$, $\forall i$ since $\Phi G (\Phi G)^T$ is positive definite. The solution to $\min\limits_{\beta} \rho(I - \beta \Phi G (\Phi G)^T)$ corresponds to the choice (20) and convergent rate

$$\begin{aligned} &\frac{\overline{\sigma}(\Phi G (\Phi G)^T) + \underline{\sigma}(\Phi G (\Phi G)^T)}{\overline{\sigma}(\Phi G (\Phi G)^T) - \underline{\sigma}(\Phi G (\Phi G)^T)} = \\ &\qquad 1 + \frac{2\underline{\sigma}(\Phi G (\Phi G)^T)}{\overline{\sigma}(\Phi G (\Phi G)^T) - \underline{\sigma}(\Phi G (\Phi G)^T)} \end{aligned} \tag{22}$$

Application of Theorem 2 guarantees that each point removed from $\mathbf{y}_r$ increases $\underline{\sigma}(\Phi G (\Phi G)^T)$ and reduces $\overline{\sigma}(\Phi G (\Phi G)^T)$. Hence the convergence rate increases. □

The convergence can be further analyzed by diagonalizing the matrix relationship $\Phi \mathbf{e}_{k+1} = (I - \beta \Phi G (\Phi G)^T)\Phi \mathbf{e}_k$

$$\Phi \mathbf{e}_{k+1} = V \text{diag}\left\{1 - \beta \sigma_i(\Phi G (\Phi G)^T)\right\} V^{-1} \Phi \mathbf{e}_k$$

where V is the eigenvector matrix of $\Phi G(\Phi G)^T$. This gives

$$V^{-1}\Phi\mathbf{e}_k = \text{diag}\left\{\left(1-\beta\sigma_i(\Phi G(\Phi G)^T)\right)^k\right\}V^{-1}\Phi\mathbf{e}_0.$$

Since $V^{-1} = V^T$, the component of $\Phi\mathbf{e}_0$ projected onto the j^{th} eigenvector of $\Phi G(\Phi G)^T$ hence evolves as $(1-\beta\sigma_i(\Phi G(\Phi G)^T))^k$. If eigenvalues i_a and i_b correspond to minimum and maximum values of $|1-\beta\sigma_i(\Phi G(\Phi G)^T)|$, then the point-to-point reference with fastest convergence rate is a scalar multiple of $\mathbf{y}_r = V_{i_a}$, and similarly the slowest is a scalar multiple of $\mathbf{y}_r = V_{i_b}$. For an arbitrary reference, the error norm sequence lies in the interval

$$\left|1-\beta\sigma_{i_b}(\Phi G(\Phi G)^T)\right|^k \le \frac{\|\Phi\mathbf{e}_k\|}{\|\Phi\mathbf{e}_0\|} \le \left|1-\beta\sigma_{i_a}(\Phi G(\Phi G)^T)\right|^k \tag{23}$$

For a given reference $\mathbf{y}_r$ the convergence rate can be most accurately estimated using the weighted sum

$$\sum_i |V_i^T\mathbf{y}_r| / \sum_i \left(|1-\beta\sigma_i(\Phi G(\Phi G)^T)||V_i^T\mathbf{y}_r|\right) \tag{24}$$

C. Robustness Margins

Theorem 4: Let there exist a multiplicative uncertainty on each element of the plant model $G(z)$, such that $\check{G}_{i,j}(z) = G_{i,j}(z)U_i(z)$. Here $\check{G}(z)$ is the actual plant and the model $G(z)$ corresponds to the matrix G used in the update law (15). A sufficient condition for monotonic convergence is that each $\arg\{U_i(e^{j\omega})\}$ lies in the open interval $(-\pi/2, \pi/2)$, demonstrating a phase margin of 90°.

Proof. This is an extension of robustness analysis for the standard gradient algorithm ($\Phi = I_{pN}$) in [13] for the SISO case. Suppose that the uncertainty can be expressed in the matrix form $\check{G} = GU$, and that point locations satisfy (13). Then from (15) the point-to-point error satisfies

$$\begin{aligned}\|\tilde{\mathbf{e}}_{k+1}\|^2 - \|\tilde{\mathbf{e}}_k\|^2 =& \beta^2\tilde{\mathbf{e}}_k^T\Phi G\check{G}^T\Phi^T(\Phi G\check{G}^T\Phi^T)^T\tilde{\mathbf{e}}_k \\ &- \beta\tilde{\mathbf{e}}_k^T\Phi G U^T G^T\Phi^T\tilde{\mathbf{e}}_k\end{aligned}$$

where $\tilde{\mathbf{e}} = \Phi\mathbf{e}$. If U is positive, the first term on the right-hand side is strictly positive for an arbitrary non-zero $\tilde{\mathbf{e}}_k$ and $\beta > 0$, and of $\mathcal{O}(\beta^2)$. Similarly the second term is of $\mathcal{O}(\beta)$ and strictly negative, and hence there always exists a $\beta > 0$ which ensures monotonic reduction in error norm. This also holds if the components of $\tilde{\mathbf{e}}_k^T\Phi G$ are reordered so that the elements corresponding to the same input are grouped, resulting in a reordering of the U matrix such that $U_{(i-1)m+i,(j-1)m+j} = U_{i,j}$. The stipulation that the components of G associated with the same input have the same uncertainty then results in U having the block diagonal structure $U = \{U_1, U_2 \ldots U_m\}$ where U_i corresponds to the i^{th} input. A sufficient condition for U to be positive definite is that each U_i is positive definite. This is the same as that arrived in [13] which goes on to show that a sufficient condition is that each $U_i(z)$ is positive-real. Note that any gain uncertainty $|U_i(e^{j\omega})|$ can be tolerated through use of a sufficiently small β. □

D. Input Energy

Theorem 5: Consider the system (1) and a point-to-point reference $\mathbf{y}_r$. ILC algorithm (15) converges to the minimum input energy solution that tracks $\mathbf{y}_r$. Furthermore, this solution is bounded by

$$\|\mathbf{u}\| \le \frac{\|\mathbf{y}_r\|}{\underline{\sigma}\left(\Phi G(\Phi G)^T\right)} \tag{25}$$

whose right-hand side strictly reduces as the number of points M is reduced.

Proof. Repeated application of (15), with $\mathbf{u}_0 = \mathbf{0}$, yields

$$\mathbf{u}_{k+1} = \left(\sum_{i=0}^{k}\left(I-(\Phi G)^T\Phi G\right)^i\right)(\Phi G)^T\mathbf{y}_r$$

which has the limit

$$\lim_{k\to\infty}\mathbf{u}_k = \left((\Phi G)^T\Phi G\right)^{-1}(\Phi G)^T\mathbf{y}_r = (\Phi G)^\dagger\,\mathbf{y}_r \tag{26}$$

This can be shown to equate to the solution to the problem

$$\min_{\mathbf{u}}\ \|\mathbf{u}\| \ \text{ subject to } \ \Phi G\mathbf{u} = \mathbf{y}_r$$

and (25) follows from the relationship

$$\left\|(\Phi G)^\dagger\right\| = \overline{\sigma}\left((\Phi G)^\dagger\right) = \frac{1}{\underline{\sigma}\left(\Phi G(\Phi G)^T\right)}.$$

It follows that the input norm is small when point-to-point locations are selected which maximize the smallest eigenvalue of $\Phi G(\Phi G)^T$. Applying Theorem 2 means that $\underline{\sigma}\left(\Phi G(\Phi G)^T\right)$ increases as each point-to-point location is removed, and hence the right-hand side of (25) reduces. □

Remark 4: For the linear time-invariant case, if only a single point is specified for each output then (25) becomes

$$\|\mathbf{u}\| \le \frac{\|\mathbf{y}_r\|}{\min\limits_{j=1,\ldots,p}\sum_{q=1}^{m}\|G_{j,q}(z)\|} \tag{27}$$

This is also achieved if the time between point locations exceeds the time taken for the impulse response to approximately go to zero (assuming asymptotic stability). □

Theorem 5 shows the benefit obtained compared with the bound $\|\mathbf{u}\| = \|G^{-1}\mathbf{y}_r\|$ corresponding to standard ILC (if it exists). This benefit increases as the number of tracked points is reduced, or their temporal spacing is increased.

IV. MULTIVARIABLE TEST FACILITY

The performance advantages established in the previous section are now confirmed experimentally using a two input, two output multivariable test facility. This system employs two interconnected differential gearboxes, together with spring-mass-damper components and is driven by two induction motors [14]. A high degree of coupling between inputs and outputs is possible which can be varied by the operator. The completed system is shown in Fig. 1 with component details listed in Table I. In this experiment, the coupling shaft is clamped, therefore the interaction of the MIMO system can be considered as zero. Modeling of the system has been achieved through open-loop frequency response tests, in which sinusoids were injected over the range of interest. The resulting output power spectra were used to generate Bode plots which have been optimally fitted by the transfer-functions

$$H_{B1} = \frac{130000.39(s+2.75)}{s(s+12.58)(s+14.20)(s+13.08)(s+15.10)} \quad (28)$$

$$H_{C2} = \frac{50000(s+1.99)}{s(s+8.41)(s+9.53)(s+10.62)(s+12.60)} \quad (29)$$

which appear in the transfer-function matrix

$$\begin{bmatrix}\phi_1\\ \phi_2\end{bmatrix} = \begin{bmatrix}H_{B1} & 0\\ 0 & H_{C2}\end{bmatrix}\begin{bmatrix}U_1\\ U_2\end{bmatrix} \quad (30)$$

where U_1 and U_2 are the voltage signals applied to each inverter. The encoder outputs (rad) from H_{B1}, H_{C2} are ϕ_1 and ϕ_2 respectively and are measured by two encoders. A proportional feedback loop has been implemented as a prestabilizer around each transfer-function, using a gain of 0.1. The closed-loop system constitutes the controlled system and, following discretization, is converted to matrix G using (3). The standard gradient method has first been applied to this system with the sinusoid reference components shown in Fig. 2. Here the period of both sinusoids is 2.5 seconds and the sampling time is $T_s = \frac{1}{100}$. It is required that each point is tracked, leading to $M = N$, $\Phi = I$ and $\mathbf{y}_r = [r_1(0)\ r_2(0)\ ...\ r_1(N-1)\ r_2(N-1)]^T$. The ILC controller update is given by (15) and an optimal β is calculated using (20). After 100 iterations the output of the system tracks the reference accurately. The error norm for this case is shown in Fig. 3. The initial error is about 110 but after approximately 20 iterations the error reduces significantly and approaches a final error norm of approximately 7. Using (24) gives the predicted convergence rate of 2.02 which matches the experimental result well in the first 6 trials but after this the effect of model uncertainty and noise then degrades the performance achieved in practice.

TABLE I

DEFINED COMPONENTS FOR THE MIMO SYSTEM

System components			
1	Encoder	2	Induction motor
3	DC motor	4	Damper
5	Spring	6	Differential gearbox
7	Coupling shaft	8	DC motor controller
9	Inverters and circuit breakers	10	Mass

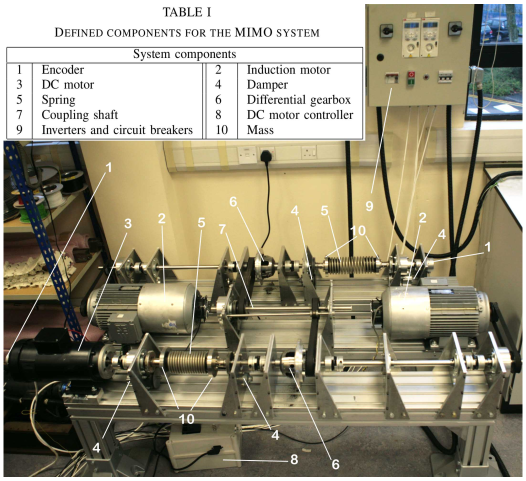

Fig. 1: Multivariable test facility with principle components labeled.

To compare performance against point-to-point ILC, only a subset of reference points have been selected to track. For output 1 these are given by $\mathbf{y}_{r,1}$ = [5.07 5.99 3.53 -2.21 -5.89] at the time points P_1 = [0.4 0.6 1 1.4 1.8] and for output 2 $\mathbf{y}_{r,2}$ = [6.75 7.24 1.00 -6.16 -7.61] at the time points P_2 = [0.4 0.8 1.2 1.6 2]. The reference $\mathbf{y}_r$ is assembled according to Section II, together with the $10 \times N$ matrix Φ. After applying algorithm (15) with optimal β (20) the results are shown in Fig. 2. The outputs of this case clearly differ to the output of the standard case since it does not need to track every single point. The starting error for this case is much smaller also since the error norm is calculated from the difference between reference points and the output at the chosen time. The starting error norm is 17.8 and after 5 iterations the error norm reaches 0.5 and continues reducing as the number of trials increases. After 100 iterations the final error norm is approximately 0.17. From (24) the predicted convergence rate for this case is 3.41 which is higher than the standard case. The eigenvectors of $\Phi G(\Phi G)^T$ are $V =$

$$\begin{bmatrix}
0.72 & 0 & 0 & 0.10 & 0.26 & 0 & 0 & 0.53 & 0 & -0.35\\
0 & -0.70 & 0.49 & 0 & 0 & -0.38 & -0.31 & 0 & 0.19 & 0\\
-0.68 & 0 & 0 & 0.01 & 0.16 & 0 & 0 & 0.55 & 0 & -0.46\\
0 & 0.54 & -0.07 & 0 & 0 & -0.35 & -0.62 & 0 & 0.43 & 0\\
0.12 & 0 & 0 & -0.51 & -0.71 & 0 & 0 & -0.03 & 0 & -0.48\\
0 & -0.37 & -0.43 & 0 & 0 & 0.56 & -0.25 & 0 & 0.52 & 0\\
0 & 0 & 0 & 0.76 & -0.18 & 0 & 0 & -0.38 & 0 & -0.50\\
0 & 0.27 & 0.64 & 0 & 0 & 0.31 & 0.36 & 0 & 0.54 & 0\\
0 & 0 & 0 & -0.40 & 0.61 & 0 & 0 & -0.53 & 0 & -0.43\\
0 & -0.12 & -0.41 & 0 & 0 & -0.53 & 0.57 & 0 & 0.46 & 0
\end{bmatrix}$$

and have corresponding eigenvalues 0.0018, 0.0025, 0.0033, 0.0036, 0.0044, 0.0045, 0.0062, 0.0069, 0.0124, 0.0131. Inserting these in $|1-\beta\sigma_i(\Phi G(\Phi G)^T)|^{-1}$ reveals convergence

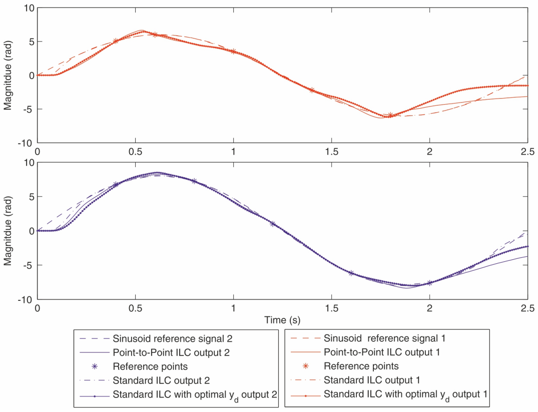

Fig. 2: System outputs using standard ILC, standard ILC with optimal $\mathbf{y}_d$ and point-to-point ILC.

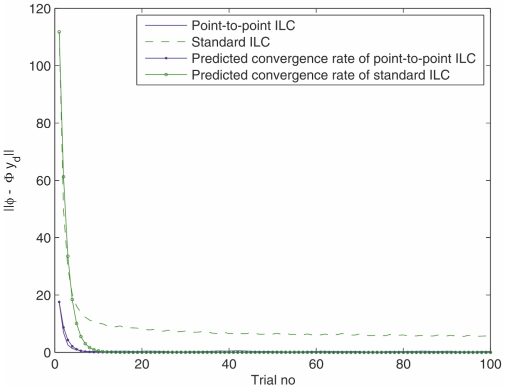

Fig. 3: Error norm for standard and point-to-point ILC.

rates of 1.32, 1.51, 1.80, 1.94, 2.46, 2.48, 5.82, 12.79, 1.50, 1.32. From (23) this hence gives a more accurate worst case convergence rate of 1.32 associated with the modes 1 and 10. The control effort norms are shown in Fig. 4 and confirm that point-to-point ILC requires a smaller control effort than standard ILC.

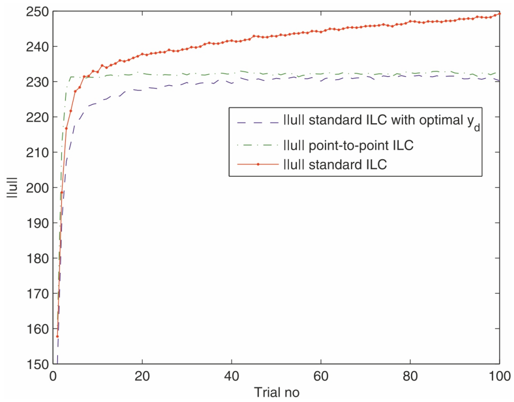

Fig. 4: Control input norm for standard ILC, standard ILC with optimal $\mathbf{y}_d$ and point-to-point ILC.

The utility of point-to-point ILC to increase the convergence speed and reduce control effort has been confirmed but it is possible that standard ILC with a reference chosen to solve (14) achieves similar results. To show the benefit of using experimental data, rather than the nominal plant model, to achieve tracking tasks, standard ILC is again applied but now with a reference that achieves the point-to-point task using the minimum input norm. This is calculated the experimental test, using the plant model and is given by

$$\mathbf{y}_d = G(\Phi G)^{\dagger}\mathbf{y}_r \tag{31}$$

Fig. 2 compares the final output of standard ILC using this reference with that of point-to-point ILC. Although they are similar, they differ since (31) is calculated using the plant model, whereas point-to-point ILC uses experimental data to solve (14). Fig. 5 shows the error norms of the two methods, and predicted convergence rates are also presented. The convergence rate of point-to-point ILC is again faster compared to standard ILC with optimal $\mathbf{y}_d$. Indeed after 10 trials point-to-point ILC reaches 0.2 while the standard ILC with optimal $\mathbf{y}_d$ error norm is 1.7. After 40 trials both methods reach the minimum error norm of approximately 0.17. The predicted convergence rate (22) approximately matches the point-to-point ILC results but it is quite different for the standard ILC with optimal $\mathbf{y}_d$ since it is clear that this method needs to track unnecessary points which causes learning transients which are magnified by noise and plant model uncertainty. Fig. 4 shows the controller norm for point-to-point ILC is lower than that of standard ILC with optimal $\mathbf{y}_d$. The point-to-point ILC controller norm reaches the maximum norm in 3 trials and keeps this value over remaining trials, whereas standard ILC with optimal reference (31) takes substantially more trials to attain this value. The experimental results hence confirm the theoretical findings that point-to-point ILC yields superior performance with a smaller controller norm.

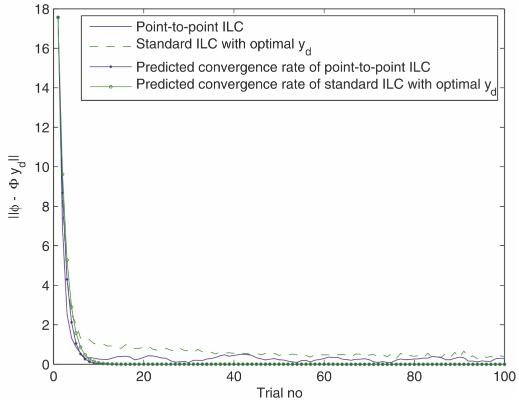

Fig. 5: Error norm and rate of convergence.

V. CONCLUSIONS

This paper derives an algorithm to address general point-to-point motion control tasks in a framework which uses learning to attain optimal solutions in the presence of model uncertainty and noise. Experimental results using a MIMO test facility confirm the practical utility and performance of the proposed approach and illustrate the benefit gained over using the standard framework.

REFERENCES

[1] D. A. Bristow, M. Tharayil, and A. 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] H. Ding and J. Wu, "Point-to-point control for a high-acceleration positioning table via cascaded learning schemes," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 5, pp. 2735–2744, 2007.

[3] J. Park, P. H. Chang, H. S. Park, and E. Lee, "Design of learning input shaping technique for residual vibration suppression in an industrial robot," *IEEE/ASME Trans Mech*, vol. 11, no. 1, pp. 55–65, 2006.

[4] J. van de Wijdeven and O. Bosgra, "Residual vibration suppression using hankel iterative learning control," *Internation Journal of Robust and Nonlinear Control*, vol. 18, pp. 1034–1051, 2008.

[5] G. Gauthier and B. Boulet, "Robust design of terminal ILC with H_∞ mixed sensitivity approach for a thermoforming oven," *Journal of Manufacturing Science and Engineering*, p. Article ID 289391, 2008.

[6] Y. Wang and Z. Hou, "Terminal iterative learning control based station stop control of a train," *Int J Cont*, vol. 84, no. 7, pp. 1263–1274, 2011.

[7] J.-X. Xu and D. Huang, "Initial state iterative learning for final state control in motion systems," *Automatica*, vol. 44, pp. 3162–3169, 2008.

[8] J.-X. Xu, Y. Chen, T. Lee, and S. Yamamoto, "Terminal iterative learning control with an application to RTPCVD thickness control," *Automatica*, vol. 35, pp. 1535–1542, 1999.

[9] P. Lucibello, S. Panzieri, and G. Ulivib, "Repositioning control of a two-link flexible arm by learning," *Automatica*, vol. 33, no. 4, pp. 579–590, 1997.

[10] S. Mishra, U. Topcu, and M. Tomizuka, "Optimization-based constrained iterative learning control," *IEEE Transactions in Control Systems Technology*, vol. 19, no. 6, pp. 1613–1621, 2011.

[11] D. H. Owens, J. J. Hätönen, and S. Daley, "Robust monotone gradient-based discrete-time iterative learning control," *International Journal of Robust and Nonlinear Control*, vol. 19, pp. 634–661, 2009.

[12] S.-G. Hwang, "Cauchy's interlace theorem for eigenvalues of Hermitian matrices," *The American Mathematics Monthly*, vol. 111, pp. 157–159, 2004.

[13] J. J. Hätönen, "Issues of algebra and optimality in iterative learning control," Ph.D. dissertation, Department of Process and Environmental Engineering, University of Oulu, 2004.

[14] T. V. Dinh, C. Freeman, and P. Lewin, "Development of a multivariable test facility for the evaluation of iterative learning controllers," *American Control Conference*, no. Accepted, 2012.