

Generalized Norm Optimal Iterative Learning Control with Intermediate Point and Sub-interval Tracking

David H. Owens^{1,2,3} Chris T. Freeman² Bing Chu²

¹Department of Automatic Control and Systems Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, UK

²School of Electronics and Computer Science, University of Southampton, Highfield, Southampton SO17 1BJ, UK

³Advanced Robotics Division, Italian Institute of Technology, Genoa, Italy

Abstract: Norm optimal iterative learning control (NOILC) has recently been applied to iterative learning control (ILC) problems in which tracking is only required at a subset of isolated time points along the trial duration. This problem addresses the practical needs of many applications, including industrial automation, crane control, satellite positioning and motion control within a medical stroke rehabilitation context. This paper provides a substantial generalization of this framework by providing a solution to the problem of convergence at intermediate points with simultaneous tracking of subsets of outputs to reference trajectories on subintervals. This formulation enables the NOILC paradigm to tackle tasks which mix “point to point” movements with linear tracking requirements and hence substantially broadens the application domain to include automation tasks which include welding or cutting movements, or human motion control where the movement is restricted by the task to straight line and/or planar segments. A solution to the problem is presented in the framework of NOILC and inherits NOILC’s well-defined convergence properties. Design guidelines and supporting experimental results are included.

Keywords: Iterative learning control, learning control, optimization, linear systems, robotics.

1 Introduction

The objective of iterative learning control (ILC) is to follow a motion profile defined over a fixed finite time interval, $0 \leq t \leq T$, using data from previous executions of this task to suitably update the control input until perfect tracking of a defined reference, $r(t)$, $t \in [0, T]$ is achieved. Often in combination with current-trial information, the aim is to sequentially improve the performance of the operation as the trial/iteration/repetition index, k , increases. Since its formal conception 30 years ago^[1], ILC has been an area of intense research interest in both theoretical and application domains^[2,3]. A body of recent work has succeeded in removing the stipulation that tracking of a reference signal is required at all points of $[0, T]$. The framework that has been developed enforces tracking at only a subset of points $\{t_j\}_{1 \leq j \leq M}$ ^[3-7] and hence addresses applications such as production line automation, crane control, satellite positioning, and robotic “pick and place” tasks in which the system output (e.g., payload position) is only critical at a finite set of prescribed time instants. It has also been used to model human motor control^[8] and within stroke rehabilitation to assist patients performing reaching and object manipulation tasks^[9].

This paper expands the framework to tackle more general

tracking problems that encompass the requirement

- 1) that a defined output signal passes through specified points at a finite number of times $\{t_j\} \subset [0, T]$,
- 2) that, on defined subintervals $[\hat{t}_j, \hat{t}_{j+1})$, defined combinations of subsets of the outputs track required signals,
- 3) that, on defined subintervals, the output vector is unspecified but lies on a plane in the output space and,
- 4) no tracking of any kind is required elsewhere on $[0, T]$.

These tasks fit many requirements needed in automation tasks which include welding or cutting movements, or human motion control where the movement is restricted by the task to straight line and/or planar segments. This expansion of the framework also enables it to tackle a special case of the “spatial ILC” problem^[10-12] in which a path is specified in space (or more generally, as a functional constraint between outputs), but no temporal constraint is imposed on how the system moves along this path (or plane). Spatial ILC presents substantial novelty, but no framework currently exists which provides widely applicable algorithmic solutions and performance analysis. The inclusion of aspects of spatial ILC hence addresses applications requiring a predefined path (or plane) to be specified between intermediate point locations, but opens up the possibility of moving along the path (or plane) in a way that achieves some other objective (e.g., control effort minimisation).

Solutions are developed in this paper using the framework of norm optimal ILC (NOILC), which is a prominent member of the class of gradient based ILC algorithms whose convergence and robustness properties have been extensively studied by many groups^[13-15]. Appearing origi-

Regular Paper
Special Issue on Latest Advances in ILC/RLC Theory and Application

Manuscript received March 26, 2014; accepted October 28, 2014
Recommended by Guest Editor Rong-Hu Chi
© Institute of Automation, Chinese Academy of Science and Springer-Verlag Berlin Heidelberg 2015

nally in [16], NOILC embeds precise control over error and input norm evolution and has received significant attention in the ILC community^[17–19], with experimental applications including an accelerator based free electron laser^[20], multi-axis laser facility^[19] and within stroke rehabilitation^[20]. Theoretical developments to NOILC include extension to discrete time^[21], an N -iteration ahead predictive solution^[22], acceleration mechanisms^[23], the inclusion of convex input constraints^[24] and analysis of the effect of non-minimum phase zeros^[25, 26]. To address 1)–4), a substantial generalization of NOILC is required to combine a general point-to-point tracking problem with partial tracking requirements on subintervals. It is achieved in this paper using an operator formulation of the plant dynamics and the application of NOILC concepts to construct an iterative algorithm that converges to a control input signal that ensures that specified values of system outputs $y(t_j)$ or linear combinations of outputs $F_j y(t_j)$ are achieved at specified intermediate times $\{t_j\}_{1 \leq j \leq M}$, whilst simultaneously ensuring that specified, and possibly different, linear combinations $P_j y(t), 1 \leq j \leq \hat{M}$ of outputs track specified signals on (possibly different) subintervals $[\hat{t}_{j-1}, \hat{t}_j]$. Such combinations could be simple tracking tasks for a subset of the outputs or they could represent constraints on output behaviour that are necessary for effective operation. If no continuous tracking is required on an interval $[t_{q-1}, t_q]$, then this is simply described by setting $P_q = 0$. The problem therefore combines the requirement to pass through defined points with the need to track defined signals on disjoint sub-intervals. The formulation is extremely general but can be realized by implementable feedback/feedforward ILC algorithms.

This paper is structured as follows. In Section 2, the intermediate weighting problem with partial tracking requirements for continuous state space systems is modelled in a general product Hilbert space setting and applied to formulate the ILC problem in Section 3 as a NOILC problem. Experimental validation results are given in Section 4 on a non-minimum-phase electromechanical test facility and their relationship to theoretical predictions is discussed. Conclusions and future work appear in Section 5.

2 Modelling in a product Hilbert space

The problem can be interpreted as a NOILC objective with emphasis placed on rapid and accurate convergence of all (or a subset) of outputs (or linear combination of outputs) at specified intermediate times $\{t_j\}_{1 \leq j \leq M}$ with effective tracking of defined signals constructed from (linear combinations of) outputs on subintervals $[\hat{t}_{j-1}, \hat{t}_j], 1 \leq j \leq \hat{M}$. Through the introduction of parameters $P_j, Q_j, 1 \leq j \leq M$ and $\hat{Q}_j, 1 \leq j \leq \hat{M}$, the problem can hence be expressed as a combination of the tracking of a standard reference $r(t), t \in [0, T]$, and intermediate point tracking. This enables NOILC solutions to be derived using the relevant operators, signals and underlying signal spaces.

2.1 Input/output notation

Let $S(A, B, C)$ be an m -output, ℓ -input, state dimension n linear, time-invariant system of the form

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), & x(0) &= x_0 \\ y(t) &= Cx(t), & t &\in [0, T] \end{aligned} \tag{1}$$

written in the operator form

$$\begin{aligned} y &= Gu + d, & G &: L_2^\ell[0, T] \rightarrow L_2^m[0, T], \\ y, d &\in L_2^m[0, T], u \in L_2^\ell[0, T] \end{aligned} \tag{2}$$

where convolution operator G and signal d are defined by

$$(Gu)(t) = \int_0^t Ce^{A(t-s)}Bu(s)ds, \quad d(t) = Ce^{At}x_0, \quad t \in [0, T] \tag{3}$$

and $T < \infty$ is fixed. Note that all functions in the range of G are continuous and differentiable.

Let $0 = t_0 < t_1 < \dots < T_M = T$ (respectively $0 = \hat{t}_0 < \hat{t}_1 < \dots < \hat{T}_{\hat{M}} = T$) be M (respectively \hat{M}) distinct points in $[0, T]$ and, for any $f \in L_2^m[0, T]$, consider the linear map $f \mapsto f^e$

$$\begin{aligned} f^e &= \begin{bmatrix} F_1 f(t_1) \\ \vdots \\ F_M f(t_M) \\ Pf \end{bmatrix} \in R^{f_1} \times \dots \times R^{f_1} \times L_2^{p_1}[0, \hat{t}_1] \times \dots \\ &\quad \times L_2^{p_{\hat{M}}}[\hat{t}_{\hat{M}-1}, T] \end{aligned} \tag{4}$$

where F_j is $f_j \times m$ and of full row rank, $1 \leq j \leq M$. The inclusion of F_j is a generalization that enables only selected elements or linear combinations of elements of f that are important in the problem to be specified at $t = t_j$.

The operator P is defined by the functional relations:

$$Pf = \begin{bmatrix} (Pf)_1 \\ \vdots \\ (Pf)_{\hat{M}} \end{bmatrix}, \quad (Pf)_j \in L_2^{p_j}[\hat{t}_{j-1}, \hat{t}_j], \quad 1 \leq j \leq \hat{M}$$

where

$$(Pf)_j(t) = P_j f(t), \quad t \in [\hat{t}_{j-1}, \hat{t}_j], \quad 1 \leq j \leq \hat{M}$$

and either

1) each P_j is a $p_j \times m$ matrix of full row rank (representing the tracking requirement $P_j y(t) \equiv r_j^P(t)$ on $[\hat{t}_{j-1}, \hat{t}_j]$)

2) or $P_j = 0$ (no tracking requirement on that interval).

They can be regarded as “partial projections” of the signal f on the domain $[\hat{t}_{j-1}, \hat{t}_j]$ representing combinations of f of control interest in these intervals.

With this notation, the “extended output” y^e from the plant is defined to be a “partially projected” output signal $(Py)(t)$ plus the values $F_j y(t_j), 1 \leq j \leq M$, at intermediate

points, i.e., dynamics can be modelled by the form

$$y^e = G_e u + d_e, \quad G_e u = (Gu)^e = \begin{bmatrix} G_1 u \\ \vdots \\ G_M u \\ PGu \end{bmatrix}, \quad d_e = \begin{bmatrix} F_1 d(t_1) \\ \vdots \\ F_M d(t_M) \\ Pd \end{bmatrix} \tag{5}$$

with $G_e : L_2^\ell[0, T] \rightarrow R^{f_1} \times \dots \times R^{f_M} \times L_2^{p_1}[0, \hat{t}_1] \times \dots \times L_2^{p_M}[\hat{t}_{M-1}, \hat{t}_M]$ as a linear bounded operator.

A technical note: If G_e does not have range dense in $L_2^{p_1}[0, \hat{t}_1] \times \dots \times L_2^{p_M}[\hat{t}_{M-1}, \hat{t}_M]$, then $L_2^{p_1}[0, \hat{t}_1] \times \dots \times L_2^{p_M}[\hat{t}_{M-1}, \hat{t}_M]$ is replaced by the closure of the range of G_e as the output Hilbert space with the natural inherited topology. This has no effect on the theory which incorporates this easily by using the virtual reference $r \in L_2^m[0, T]$.

Here, $G_j : L_2^\ell[0, T] \rightarrow R^{f_j}$ is defined by the relation

$$G_j u = F_j \int_0^{t_j} C e^{A(t_j-t)} B u(t) dt \tag{6}$$

and $(PG)_j : L_2^\ell[0, T] \rightarrow L_2^{p_j}[\hat{t}_{j-1}, \hat{t}_j]$ is defined by

$$((PG)_j u)(t) = P_j \int_0^t C e^{A(t-t')} B u(t') dt', \quad t \in [\hat{t}_{j-1}, \hat{t}_j]. \tag{7}$$

A reference signal r^e is constructed by specifying the desired values of $F_j y(t_j), 1 \leq j \leq M$ and the desired signals $r_j^P(t)$ for $P_j y(t), t \in [\hat{t}_{j-1}, \hat{t}_j], 1 \leq j \leq \hat{M}$. The attainability of this reference by the system requires consistency between these characterizations, and is addressed as follows.

Tracking signal characterization: It is assumed that r^e can be generated using a (possibly non-unique, but continuous) virtual reference r for the plant output y such that $r - d$ lies in the range of G . In this case, the tracking error $e^e = r^e - y^e$ can be associated with the ‘‘usual’’ tracking error $e(t) = r(t) - y(t)$ used in NOILC as

$$e^e(t) = r^e - y^e = (r - y)^e = \begin{bmatrix} F_1 e(t_1) \\ \vdots \\ F_M e(t_M) \\ P e \end{bmatrix}. \tag{8}$$

It is useful to note the following:

1) The virtual reference r and the associated virtual error e will play a role in the simplification of the theory and relating the optimization problems used in the algorithm to, more traditional, optimal control problems. Note that neither r nor e are needed in the final computations as they are always multiplied by operations that convert them into components of the extended signals such as $P_j e$ that can be computed from the specified signal $P_j r$ and the measured plant output.

2) The pure intermediate point problem is obtained by removing Pf from the definition of f^e whilst NOILC is obtained by removing the intermediate points and looking at

the case of $\hat{M} = 1, \hat{t}_1 = T$ and $P = I$. In terms of defining f^e , this is just (respectively)

$$f^e = \begin{bmatrix} F_1 f(t_1) \\ \vdots \\ F_M f(t_M) \end{bmatrix} \in R^{f_1} \times \dots \times R^{f_M}, \quad f^e = f \in L_2^m[0, T]. \tag{9}$$

It is noted that the first is obtained in what follows by setting $P_j = 0, 1 \leq j \leq \hat{M}$ whilst the second is obtained using $F_j = 0, 1 \leq j \leq M$ and $\hat{Q}_j = Q, 1 \leq j \leq \hat{M}$ and $P_j = I$.

3) The extension of the problem to the case of all $t_j < T$ is achieved by increasing M by one and then defining an additional point at $t_M = T$ with the corresponding $F_M = 0$.

4) If there are no tracking requirements on some subinterval $[\hat{t}_{q-1}, \hat{t}_q]$, then simply set $P_q = 0$.

To illustrate the wide range of tracking signals that the construction allows, consider the following example of a reference signal for the case of $m = 3$ and control over a time interval $[0, 12]$:

1) The intermediate points are defined by the requirement that output $y_1(t)$ passes through the points $y_1 = 1, 2, 3, 2, 1, 0$ at times $t = 2, 4, 6, 8, 10, 12$ respectively.

2) Dynamics of all outputs are constrained during the subinterval $[4, 8] \subset [0, 12]$, to satisfy the planar/spatial constraint $y_1(t) + y_2(t) + y_3(t) \equiv 2$.

The resultant data set is $t_1 = 2, t_2 = 4, t_3 = 6, t_4 = 8, t_5 = 10, t_6 = 12$ with $M = 6$ and $F_1 = F_2 = \dots = F_6 = [1, 0, 0]$. The remaining data sets are $\hat{t}_1 = 4, \hat{t}_2 = 8, \hat{t}_3 = 12$ with $\hat{M} = 3$. Hence, $P_1 = P_3 = 0$ and $P_2 = [1, 1, 1]$ with reference signals $r_1^P(t) \equiv 0, r^P(t) \equiv 0$ and $r_2^P(t) \equiv 2$ on the intervals $[0, 4), [4, 8], (8, 12]$.

2.2 Hilbert space structures and operators

The product space $R^{f_1} \times \dots \times R^{f_M} \times L_2^{p_1}[0, \hat{t}_1] \times \dots \times L_2^{p_M}[\hat{t}_{M-1}, T]$ is a Hilbert space with inner product

$$\langle (v_1, \dots, v_M, f_1, \dots, f_M), (w_1, \dots, w_M, g_1, \dots, g_M) \rangle_{[Q]} = \sum_{j=1}^M v_j^T Q_j w_j + \sum_{j=1}^{\hat{M}} \int_{\hat{t}_{j-1}}^{\hat{t}_j} f_j(t)^T \hat{Q}_j g_j(t) dt \tag{10}$$

where the $f_j \times f_j$ matrices $Q_j, 1 \leq j \leq M$ and the $p_j \times p_j$ matrices $\hat{Q}_j, 1 \leq j \leq \hat{M}$ are all symmetric and positive definite. The simple notation $[Q]$ is used to denote the data set $[Q] = \{Q_1, \dots, Q_M, \hat{Q}_1, \dots, \hat{Q}_{\hat{M}}\}$.

Using the normal definition of the associated norm, namely $\|q\| = \sqrt{\langle q, q \rangle}$, the squared norm of the error e^e , expressed in terms of the ‘‘underlying’’ error e , is simply

$$\|e^e\|_{[Q]}^2 = \sum_{j=1}^M e^T(t_j) F_j^T Q_j F_j e(t_j) + \sum_{j=1}^{\hat{M}} \int_{\hat{t}_{j-1}}^{\hat{t}_j} e^T(t) P_j^T \hat{Q}_j P_j e(t) dt. \tag{11}$$

Finally, note that

1) $L_2^\ell[0, T]$ is also a Hilbert space with inner product and norm defined by

$$\begin{aligned} \langle u_1, u_2 \rangle_R &= \int_0^T u_1^T(t) R u_2(t) dt \\ \|u\|_R^2 &= \int_0^T u^T(t) R u(t) dt \end{aligned} \tag{12}$$

where $\ell \times \ell$ matrix R is symmetric and positive definite.

2) Equation (11) suggests that any objective function containing the term

$$\sum_{j=1}^M e^T(t_j) H_j e(t_j) + \sum_{j=1}^{\hat{M}} \int_{\hat{t}_{j-1}}^{\hat{t}_j} e^T(t) Q(t) e(t) dt$$

with $Q(t)$ symmetric, positive semi-definite and piece-wise constant on $[0, T]$ and each H_j symmetric and positive semi-definite, can be associated with an ILC problem of the type discussed in this paper by identifying the intervals $(\hat{t}_{j-1}, \hat{t}_j), 1 \leq j \leq \hat{M}$ where either $Q(t) = 0$ or $Q(t)$ is constant with rank $p_j \leq m$ and setting

$$\begin{aligned} Q(t) &= P_j^T \hat{Q}_j P_j, \quad t \in [\hat{t}_{j-1}, \hat{t}_j], \hat{Q}_j = \hat{Q}_j^T > 0, \quad 1 \leq j \leq \hat{M} \\ H_j &= F_j^T Q_j F_j, \quad Q_j = Q_j^T > 0, \quad 1 \leq j \leq \hat{M} \end{aligned} \tag{13}$$

where each F_j has full row rank and either P_j has full row rank or is zero.

3 Formulation of the ILC problem

In its original form, NOILC for linear, time-invariant, state-space systems is based on the minimization of an objective function

$$\begin{aligned} u_{k+1} &= \arg \min \{ J(u_{k+1}) = \|e_{k+1}\|^2 + \|u_{k+1} - u_k\|^2 : \\ &e = r - y, y = Gu + d \}, k \geq 0. \end{aligned} \tag{14}$$

It is clear that the new mixed problem introduced in the previous section can be expressed in this form by identifying G with G_e and the norms with the new norms defined. It follows that the iterative procedure defined by constructing u_{k+1} on iteration $k + 1$ by minimizing the objective function

$$\begin{aligned} J(u_{k+1}) &= \sum_{j=1}^M e_{k+1}^T(t_j) F_j^T Q_j F_j e_{k+1}(t_j) + \\ &\sum_{j=1}^{\hat{M}} \int_{\hat{t}_{j-1}}^{\hat{t}_j} e_{k+1}^T(t) Q(t) e_{k+1}(t) dt + \\ &\int_0^T (u_{k+1}(t) - u_k(t))^T R (u_{k+1}(t) - u_k(t)) dt \\ Q(t) &= P_j^T \hat{Q}_j P_j, \quad t \in (\hat{t}_{j-1}, \hat{t}_j], \quad 1 \leq j \leq M \end{aligned} \tag{15}$$

subject to the state space dynamics of $S(A, B, C)$ is precisely a NOILC iteration for G_e . Note that the criterion is expressed in terms of the virtual error e_{k+1} rather than the extended error e_{k+1}^e .

In the next sections, the solution is found in its Hilbert space setting and then formulated as the solution of a two-point boundary value problem with M ‘‘jump’’ conditions.

3.1 Formal solution of the intermediate point algorithm with subinterval weighting

The new algorithm can be written, in the notation of the previous section, as

$$\begin{aligned} u_{k+1} &= \arg \min \{ J(u_{k+1}) = \|e_{k+1}^e\|_{[Q]}^2 + \|u_{k+1} - u_k\|_R^2 : \\ &e^e = r^e - y^e, y^e = G_e u + d_e \}. \end{aligned} \tag{16}$$

This has identical structure to the NOILC problem originally proposed in [16], the only differences being the definition of operators, signals and the underlying Hilbert spaces. The iterative solution to (16) is hence written formally as

$$u_{k+1} = u_k + G_e^* e_{k+1}^e \Rightarrow e_{k+1}^e = (I + G_e G_e^*)^{-1} e_k^e \tag{17}$$

where G_e^* is the adjoint operator of G_e in the topology introduced in Section 2. This yields the monotonicity and convergence properties summarised in the following theorem. Note that this guarantees monotonic convergence of $\|e_k^e\|_{[Q]}$ for all possible choices of symmetric positive definite $R, [Q]$, and furthermore that $\|e_k^e\|_{[Q]} \rightarrow 0$ implies $\|e_k^e\|_R \rightarrow 0$.

Theorem 1. Under the conditions defined above and given an initial choice of input u_0 , the extended error norm is monotonically decreasing

$$\|e_{k+1}^e\|_{[Q]} \leq \|e_k^e\|_{[Q]}, \quad \forall k \geq 0 \tag{18}$$

with equality holding if and only if $u_{k+1} = u_k$. As a consequence, there exists a real number $E_\infty \geq 0$ such that

$$\lim_{k \rightarrow \infty} \|e_k^e\|_{[Q]} = E_\infty \tag{19}$$

and $E_\infty = 0$ if and only if

$$\lim_{k \rightarrow \infty} e_k^e = 0. \tag{20}$$

This condition is satisfied if the extended reference r^e lies in the closure of the range of G_e . Finally,

1) the resultant input sequence satisfies

$$\sum_{k=1}^{\infty} \|u_{k+1} - u_k\|_R^2 < \infty \tag{21}$$

2) if the kernel of G_e^* is $\{0\}$, the initial input $u_0 = 0$ and the signal $r^e - d_e$ lies in the range of $G_e G_e^*$, then the input sequence $\{u_k\}_{k \geq 0}$ converges to the unique stationary point, u_∞ , of the Lagrangian associated with the optimization problem

$$u_\infty = \arg \min \{ J(u) = \|u\|^2 : r^e = G_e u + d_e \}. \tag{22}$$

In other word, it converges to the input achieving the tracking requirement that has the smallest energy as represented by its norm in \mathcal{U} .

Proof. The result follows from the work in [16] apart from the last statement. To show this, note that the Lagrangian associated with the minimum energy problem is just, with Lagrange multiplier $\lambda \in \mathcal{R}_e$,

$$\mathcal{L}(u, \lambda) = \|u\|_{\mathcal{U}}^2 + 2 \langle \lambda, r^e - G_e u - d_e \rangle_{\mathcal{R}_e} \tag{23}$$

which has a stationary point when $u_\infty = G_e^* \lambda$ and $r^e = G_e u_\infty + d_e$, i.e., $r^e - d_e = G_e G_e^* \lambda$. The stationary point solution for λ exists and is unique as is the defined input solution u_∞ . This solution solves the minimum energy problem follows from the projection theorem in Hilbert space and the fact that the linear variety $\{u : r^e = G_e u + d_e\}$ is non-empty and closed. Next, we write the iterations for the ILC algorithm in the form

$$u_{k+1} = u_k + G_e^* e_{k+1}, \quad e_{k+1} = (I + G_e G_e^*)^{-1} e_k^e, \quad k \geq 0$$

$$\lim_{k \rightarrow \infty} e_k^e = 0. \tag{24}$$

Using the notation $X = (I + G_e G_e^*)^{-1}$, implies that $u_k = G_e^* \sum_{j=1}^k X^j e_0^e$. Writing $r^e - d_e = e_0^e = G_e G_e^* \lambda_0$ and noting that $X G_e G_e^* = I - X$ gives

$$u_k = G_e^* \left(\sum_{j=1}^k X^j e_0^e \right) = G_e^* \left(\sum_{j=0}^{(k-1)} X^j (I - X) \lambda_0 \right) = G_e^* (I - X^k) \lambda_0. \tag{25}$$

It follows that u_k converges in norm to an input \hat{u}_∞ satisfying $\hat{u}_\infty = G_e^* \lambda_0$ and $r^e = G_e \hat{u}_\infty + d_e$, i.e., it converges to a stationary point of the Lagrangian which is unique. Clearly $\hat{u}_\infty = u_\infty$ and $\lambda_0 = \lambda$. \square

It is useful to note the following observations:

1) Convergence to the minimum energy controller distinguishes the proposed solution from any solution computed using NOILC and an arbitrarily chosen reference signal $r(t)$ defined on the whole time interval $[0, T]$ but satisfying the intermediate point and partial tracking requirements.

2) If all $P_j = 0$, the output space is finite dimensional and $G_e G_e^*$ is a square matrix, self adjoint in the defined topology. The approach then reduces to the intermediate point tracking problem^[5]. In addition, G is right-invertible, then the convergence is geometric as

$$\|e_{k+1}^e\|_{[Q]} \leq \frac{1}{1 + \underline{\sigma}^2} \|e_k^e\|_{[Q]}, \quad \forall k \geq 0 \tag{26}$$

where $\underline{\sigma}^2$ is the smallest eigenvalue of the matrix $G_e G_e^*$. In this case, invertibility of $G_e G_e^*$ ensures convergence to zero error.

3) In addition, taking the case of $f_j = m$ and $F_j = I_m$ for example, the choice of Q_j is available to the designer to influence the nature and form of the convergence, e.g., choosing $Q_j = \lambda^{j-1} Q_0$ for some symmetric positive definite matrix Q_0 and some choice of $\lambda \in (0, 1)$ will prioritise the reduction of errors at t_{j-1} before t_j .

4) In a similar manner, choosing different values for \hat{Q}_j , $1 \leq j \leq \hat{M}$ places different priorities on convergence on the subintervals $[\hat{t}_j, \hat{t}_{j+1})$.

In the discussion that follows, it is assumed that at least one \hat{Q}_j is non-zero.

3.2 The two-point boundary value problem on iteration $k + 1$:

The solution (17) of NOILC problem (16) requires operator G_e^* , which is computed in the Appendix. Substituting

this operator into (17) converts the solution, u_{k+1} , into the form of a two-point boundary value problem (TPBVP) with jump conditions, given by

$$\begin{aligned} \dot{x}_{k+1}(t) &= Ax_{k+1}(t) + Bu_{k+1}(t) \\ x_{k+1}(0) &= x_0 \\ u_{k+1}(t) &= u_k(t) + R^{-1} B^T p(t) \\ \dot{p}(t) &= -A^T p(t) - C^T Q(t) e_{k+1}(t), \quad t \in [0, T] \end{aligned} \tag{27}$$

where

$$\begin{aligned} p(T) &= C^T F_M^T Q_M F_M e_{k+1}(T) \\ p(t_j-) &= p(t_j+) + C^T F_j^T Q_j F_j e_{k+1}(t_j), \quad 1 \leq j < M. \end{aligned}$$

The next subsection converts this characterization into a feedback/feedforward realization using Riccati techniques.

3.3 A causal algorithm

The formal solution as a boundary value problem can be expressed either as a causal feedforward solution when $Q(t) = 0^{[5]}$ or more generally, as a causal feedback plus feedforward solution. The details are described below.

The approach to constructing a state feedback plus predictive/feedforward term causal solution is familiar and closely related to the classical linear quadratic tracking problem. The solution provided below is not unique but all solutions are equivalent in the sense that they generate the same u_{k+1} . The approach is illustrated by writing

$$p(t) = -K(t)x_{k+1}(t) + \xi_{k+1}(t) \tag{28}$$

where $K(t)$, $\xi_{k+1}(t)$ are assumed to be continuous and differentiable in each open segment (t_j, t_{j+1}) but possibly discontinuous at the points $t = t_j$ and $K(t)$ is symmetric. The state vector is continuous on $[0, T]$.

A causal feedback solution for u_{k+1} is constructed as follows:

Step 1. Compute off-line the Riccati feedback matrix from the matrix differential equation, terminal condition and jumps

$$\begin{aligned} \dot{Q}(t) &= P_j^T \hat{Q}_j P_j, \quad t \in (\hat{t}_{j-1}, \hat{t}_j], \quad 1 \leq j \leq \hat{M} \\ \dot{K}(t) + A^T K(t) + K(t)A - K(t)BR^{-1}B^T K(t) + C^T Q(t)C &= 0 \end{aligned} \tag{29}$$

with

$$\begin{cases} K(T) = C^T F_M^T Q_M F_M C \\ K(t_j-) = K(t_j+) + C^T F_j^T Q_j F_j C, \quad 1 \leq j < M. \end{cases}$$

Step 2. On the $(k + 1)$ -th iteration, compute off-line the predictive feedforward term from the differential equation, terminal condition and jumps

$$\begin{aligned} \dot{\xi}_{k+1}(t) + A^T \xi_{k+1}(t) - K(t)BR^{-1}B^T \xi_{k+1}(t) - K(t)Bu_k(t) + C^T Q(t)r(t) &= 0 \\ \xi_{k+1}(T) &= C^T F_M^T Q_M F_M r(T) \\ \xi_{k+1}(t_j-) &= \xi_{k+1}(t_j+) + C^T F_j^T Q_j F_j r(t_j), \quad 1 \leq j < M. \end{aligned} \tag{30}$$

Although these equations contain the virtual reference $r(t)$, note that $F_j r(t_j), 1 \leq j \leq M$ is specified in the tracking problem and that the product $Q(t)r(t) = P_j^T \hat{Q}_j P_j r(t), t \in [t_{j-1}, t_j]$ is known as $P_j r(t)$ is specified on $[t_{j-1}, t_j]$.

Step 3. Implement the control law

$$u_{k+1}(t) = u_k(t) + R^{-1} B^T [\xi_{k+1}(t) - K(t)x_{k+1}(t)] \quad (31)$$

to achieve the predicted monotonic convergence.

3.4 Performance and design observations

Given the tracking objectives, the choice of $R, \hat{Q}_j, 1 \leq j \leq \hat{M}, Q_j, 1 \leq j \leq M$ and $t_j, 1 \leq j < M$ are open to the designer of the system. As with the choice of Q and R in the traditional linear quadratic regulator (LQR) problem, there are no strict rules for their selection but the following comments can be made.

1) The case when $\hat{Q}_j \neq 0$ is equivalent, intuitively, to a requirement that the output tracks reference $P_j r(t)$ on subintervals of $[0, T]$. Changing the relative values of \hat{Q}_j will change the relative convergence rates on the intervals with the general rule that, increasing \hat{Q}_q will improve convergence rates on the subinterval $[\hat{t}_{q-1}, \hat{t}_q]$. Also decreasing the values of the \hat{Q}_j relative to the Q_j will place increasing emphasis on the specific intermediate points at times $t_j, 1 \leq j \leq M$ leading to an expected faster reduction of errors at the intermediate points and slower convergence elsewhere. Details will depend upon plant dynamics and the choice of R .

2) In general terms, reducing the value of R is expected, intuitively to tend to accelerate algorithm convergence.

4 Experimental results

The approach has been tested on a six degree of freedom anthropomorphic robotic arm (see Fig. 1) whose five rotary joints are composed of PowerCubes (Schunk GmbH & Co.) incorporating brushless servomotors with integrated power electronics and transmission. These communicate with a dSPACE ds1103 control board via a CAN bus at a rate of 500 kbit/s, and each is controlled in torque mode. The task considered involves planar movement using only two actuated joints. Starting from an initial location, the end-effector moves to a specified location (e.g., to pick up a tool head), moving along a path (replicating a cutting or welding action), returning to another specified location (e.g., to replace/clean a tool head), and returning to the initial location.

The plant model has been identified using frequency response tests and is given by transfer function matrix (32). Cascaded current and velocity control loops are used, with inputs and outputs measured in radians.

The task requires the output to pass through selected points at specified times plus the additional requirement that a linear combination of outputs takes a specified functional value on a proper subinterval. More precisely, zero initial conditions are assumed for each iteration and tracking is represented by using intermediate point locations, together with an output constraint that enforces a path following component on a subinterval. The path following component does not specify output values but does specify their functional relationship.

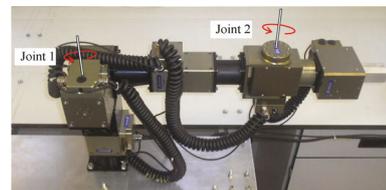


Fig. 1 Robot showing actuated joints

The intermediate output values are given by the $M = 5$ time points

$$t_1 = 3, \quad t_2 = 4, \quad t_3 = 10, \quad t_4 = 11, \quad t_5 = T = 14$$

with the corresponding outputs

$$r(t_1) = \begin{bmatrix} 0.3689 \\ 1.5480 \end{bmatrix}, \quad r(t_2) = \begin{bmatrix} 0.9570 \\ 0.3883 \end{bmatrix}, \quad r(t_3) = \begin{bmatrix} 1.5660 \\ 1.2609 \end{bmatrix},$$

$$r(t_4) = \begin{bmatrix} 0.4191 \\ 1.6723 \end{bmatrix}, \quad r(t_5) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

where $F_j = I, 1 \leq j \leq 5$. In addition, tracking is also defined by the constraint in the interval $[4, 10]$.

$$y_2 - 1.433y_1 = -0.9832 \quad (\text{i.e., } \hat{M} = 3, \hat{t}_1 = 4, \hat{t}_2 = 10, \hat{t}_3 = T),$$

$$P_1 = [0, 0], \quad P_2 = [-1.433, 1.0], \quad P_3 = [0, 0],$$

$$r_1^P(t) \equiv 0, \quad r_2^P(t) = -0.9832, \quad r_3^P(t) \equiv 0 \quad (32)$$

representing a coordinated spatial movement of the two joints in terms of relative values. Note that this does not specify numerical values of $y_1(t)$ or $y_2(t)$ individually at any point in $[4, 10]$. To achieve the task, the control action (31) has been implemented using the state feedback matrix (29) and predictive feedforward term (30). As the states are not measured directly, a Kalman filter provides the state

$$G(s) = \begin{bmatrix} \frac{3.34e^5 s^3 + 2.01e^7 s^2 + 3.87e^8 s + 2.34e^9}{s^8 + 217.8s^7 + 1.9e^4 s^6 + 8.42e^5 s^5 + 2e^7 s^4 + 2.42e^8 s^3 + 1.17e^9 s^2} & \frac{720}{s^3 + 95s^2 + 1300s + 8000} \\ \frac{600}{s^3 + 89s^2 + 1150s + 7500} & \frac{4.5e^4 s^3 + 1.71e^7 s^2 + 1.72e^9 s + 1.85^{10}}{s^9 + 203s^8 + 1.72e^4 s^7 + 7.82e^5 s^6 + 2.06e^7 s^5 + 3.14e^8 s^4 + 2.55e^9 s^3 + 8.50e^9 s^2} \end{bmatrix} \quad (33)$$

estimates using process and measurement disturbance covariance weights of $10I$ and I respectively.

To provide a comparative approach to quantify convergence rates, the uniquely defined iteration index, k^* , corresponding to the first index where the tracking criterion

$$\|e_k^e\| = \|r^e - y_k^e\| < \epsilon \|r^e\| \tag{34}$$

is met is noted. A value of $\epsilon = 0.01$ is used in all the results that follow corresponding to an improvement in tracking by a factor of 100 relative to the extended reference.

Fig. 2 shows the converged paths after 200 trials using $R = I$, $Q_j = 50I$, $1 \leq j \leq 5$ and $\hat{Q}_j = 2$, $1 \leq j \leq 3$. Corresponding output angles and control input signals are shown in Fig. 3, together with signals on trials 1, 3 and 5. Norm results over all trials are shown in Fig. 4 for a variety of Q_j and \hat{Q}_j .

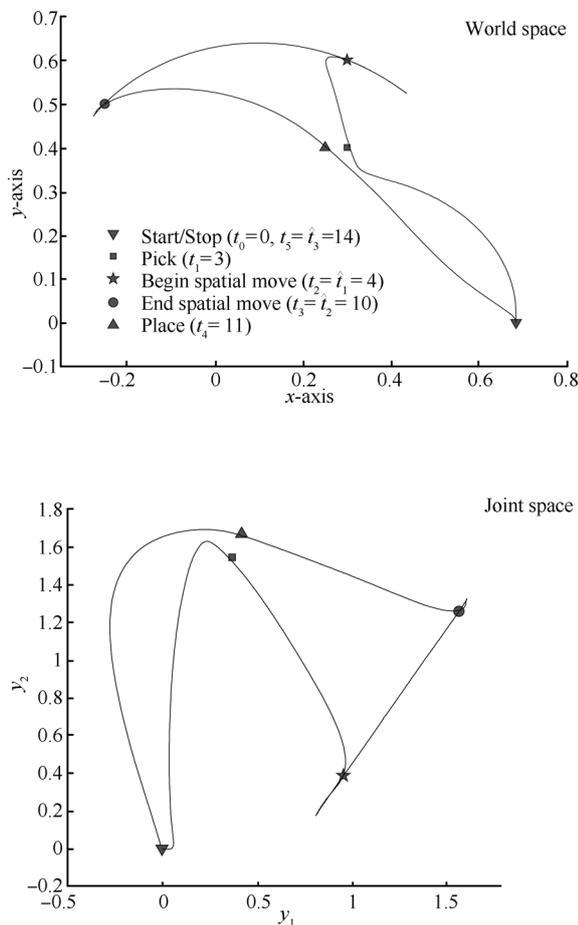


Fig. 2 Observed paths on trial 200 in world space (corresponding to path of robot end-effector in Cartesian space) and joint space, using $Q_j = 50I$ and $\hat{Q} = 2$

Table 1 shows summary data, confirming that the both the intermediate point tracking and the spatial tracking components are achieved to high accuracy. As Q_j increases more emphasis is placed on minimizing the intermediate

point error, compared to the spatial tracking error. In practice, an optimum compromise can be found between Q_j and \hat{Q}_j values which provides convergence to low values of each objective. At the same time, there is clearly a trade-off between rapid convergence to low error, and the control effort required to achieve it. Lower values of Q_j and \hat{Q}_j typically enable less energy to be expended, but k^* increases.

Table 1 Summary norm data, where $\|Pe_{k^*}\|$ is the spatial error norm, and the intermediate point (IP) tracking norm is $\|e_{k^*}^e\| - \|Pe_{k^*}\|$

Q_j	\hat{Q}_j	$\ e_{k^*}^e\ $	$\ Pe_{k^*}\ $	IP norm	$\ u_{k^*}\ $	k^*
1I	1	0.1762	0.0257	0.1505	12.911	>200
10I	1	0.0560	0.0391	0.0169	14.507	39
20I	1	0.0509	0.0422	0.0087	14.615	27
50I	1	0.0453	0.0417	0.0036	14.804	20
100I	1	0.0446	0.0433	0.0013	15.048	17
500I	1	0.0351	0.0344	0.0007	16.373	17
1000I	1	0.0346	0.0343	0.0003	17.464	19
1I	2	0.1604	0.0122	0.1482	13.025	191
10I	2	0.0455	0.0285	0.0170	14.578	33
20I	2	0.0406	0.0308	0.0098	14.678	22
50I	2	0.0343	0.0302	0.0041	14.869	13
100I	2	0.0298	0.0282	0.0016	15.109	11
500I	2	0.0244	0.0238	0.0006	16.466	10
1000I	2	0.0258	0.0255	0.0003	17.520	11
1I	10	0.1524	0.0041	0.1483	13.471	187
10I	10	0.0266	0.0109	0.0157	14.964	27
20I	10	0.0203	0.0083	0.0120	15.038	18
50I	10	0.0167	0.0107	0.0060	15.246	12
100I	10	0.0153	0.0126	0.0027	15.514	9
500I	10	0.0172	0.0165	0.0007	16.850	8
1000I	10	0.0128	0.0125	0.0003	17.892	8
1I	20	0.1518	0.0018	0.1500	13.850	189
10I	20	0.0242	0.0099	0.0143	15.386	26
20I	20	0.0198	0.0077	0.0121	15.481	18
50I	20	0.0133	0.0063	0.0070	15.683	11
100I	20	0.0111	0.0068	0.0042	15.971	10
500I	20	0.0131	0.0124	0.0007	17.342	9
1000I	20	0.0095	0.0090	0.0005	18.314	8

The combined intermediate point and spatial tracking approach is now compared to the standard ILC framework. As is shown in Theorem 1, in the absence of plant model mismatch this method converges to the minimum energy solution to the problem, and hence the most appropriate comparison is to use standard NOILC with $r(t)$ calculated in advance using the nominal plant model to achieve the intermediate point and spatial tracking tasks whilst also requiring minimum input energy. This signal is shown in Fig. 3 where it is labeled $\tilde{r}(t)$. Using this predefined reference, standard NOILC corresponds to removing the intermediate points and using $\hat{M} = 1$, $\hat{t}_1 = T$, $P = I$.

Results in Table 2 show that standard NOILC requires a significantly larger control effort norm to obtain similar levels of error, while requiring a greater number of trials.

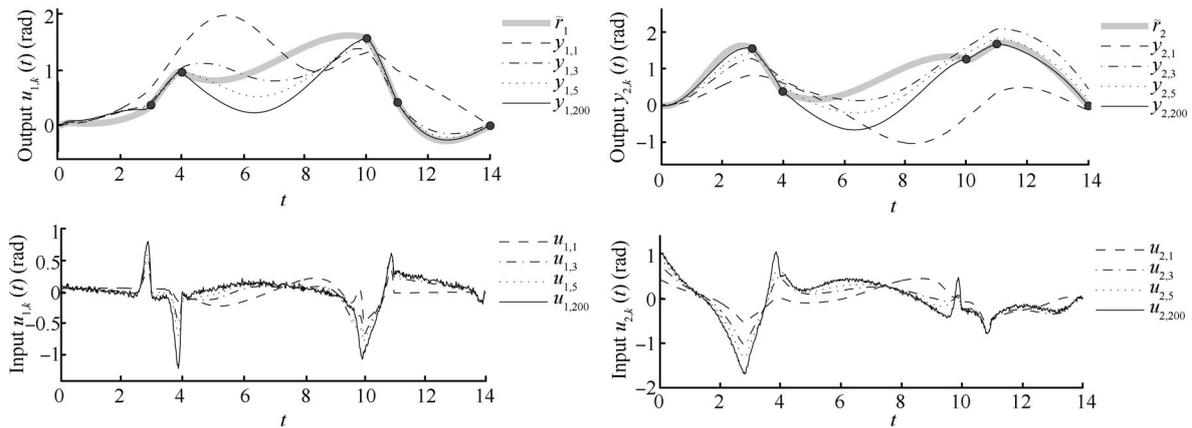


Fig. 3 Output angles, $y_{1,k}(t)$, $y_{2,k}(t)$ and corresponding control inputs $u_{1,k}(t)$, $u_{2,k}(t)$ over 200 trials, using $Q_j = 50I$, $\hat{Q} = 2$. The minimum energy solution, calculated using the nominal plant model, is shown for comparison and labeled $\tilde{r}(t)$

Table 2 Standard NOILC using a reference $r(t)$ equal to the minimum energy solution

\hat{Q}_j	$\ e_{k^*}^e\ $	$\ Pe_{k^*}\ $	IP norm	$\ u_{k^*}\ $	k^*
1	0.0745	0.0480	0.0265	15.463	52
2	0.0509	0.0343	0.0166	15.774	27
10	0.0177	0.0148	0.0029	16.315	11
20	0.0105	0.0090	0.0015	16.369	10

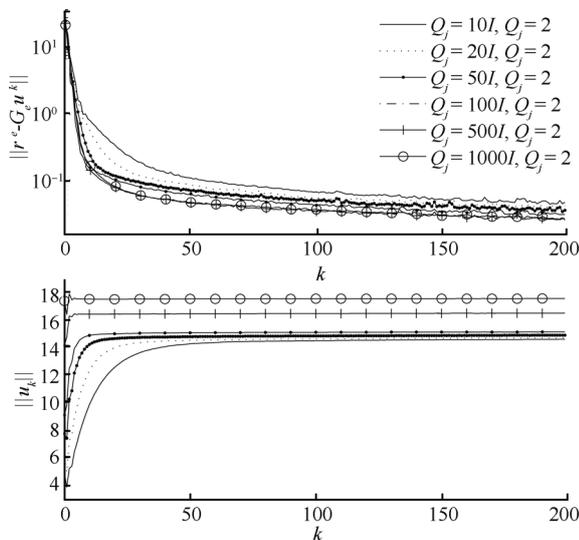


Fig. 4 Error norms (combined intermediate point tracking and spatial tracking) and input norms over 200 trials, for a variety of Q_j and \hat{Q}_j

5 Conclusions

A well-defined NOILC problem which places specific emphasis on two objectives, namely, simultaneous fast accurate tracking at a finite number of intermediate points of time within the interval $[0, T]$ plus tracking of defined linear combinations of outputs on sub-intervals of $[0, T]$, can be approached and solved using the operator methods underpinning those of NOILC in [16]. The resultant algorithm in-

herits the original convergence properties and monotonicity properties that makes NOILC so attractive. The formulation is quite general for linear time invariant state space systems and can be represented in a feedforward structure (when $Q(t) \equiv 0$), but more generally, has the form of familiar current trial Riccati state feedback and predictive/feedforward terms generated from off-line, reverse-time simulations of differential equations with defined jump conditions at the intermediate points selected. An analysis of the effect of the choice of weighting matrices provides guidelines to their selection in terms of convergence rates and the effect of removing intermediate points from the problem.

Experimental results indicate that the predictions of error monotonicity and rapid convergence can apply in practice with great accuracy and with robustness to modelling errors. Further research is needed to address more general issues of parameter choice, a formal characterization of the algorithm robustness to modelling errors and the effects of varying initial condition.

Appendix (Computation of the adjoint operator G_e^*)

At the centre of NOILC is the characterization^[16] of the adjoint operator mapping inputs into defined outputs. Therefore, purpose of this section, is to compute the relevant adjoint operator G_e^* of G_e given the Hilbert space topology defined above. By definition,

$$\langle (w_1, \dots, w_M, v_1, \dots, v_{\hat{M}}), G_e u \rangle_{[Q]} = \langle G_e^* (w_1, \dots, w_M, v_1, \dots, v_{\hat{M}}), u \rangle_R \tag{35}$$

then the adjoint is computed from adjoints of $G_j, 1 \leq j \leq M$ and (PG) .

1) Adjoint operator of G_j : First look at G_j via

$$w_j^T Q_j F_j \int_0^{t_j} C e^{A(t_j-t)} B u(t) dt = \int_0^{t_j} (R^{-1} B^T e^{A^T(t_j-t)} C^T F_j^T Q_j w_j)^T R u(t) dt \tag{36}$$

and hence deduce that

$$(G_j^* w_j)(t) = \begin{cases} R^{-1} B^T e^{A^T(t_j-t)} C^T F_j^T Q_j w_j, & 0 \leq t \leq t_j \\ 0, & t > t_j \end{cases} \tag{37}$$

which can be written in the form

$$(G_e^* w_j)(t) = R^{-1} B^T p_j(t) \tag{38}$$

where on $[0, t_j]$,

$$\begin{aligned} \dot{p}_j(t) &= -A^T p_j(t) \\ p_j(t_j-) &= C^T F_j^T Q_j w_j \end{aligned} \tag{39}$$

and $p_j(t) = 0$ on $(t_j, T]$.

2) Adjoint operator of PG : Suppose the Hilbert topology in the range of G is defined (with Q arbitrary as it plays no role in the final algorithm) by the inner product

$$\begin{aligned} \langle y, w \rangle &= \int_0^T y^T(t) Q w(t) dt \\ Q &= Q^T > 0. \end{aligned} \tag{40}$$

Firstly, the adjoint of G is computed as the map $w = G^* v$ as

$$\begin{aligned} w(t) &= R^{-1} B^T p_{M+1}(t) \\ \dot{p}_{M+1}(t) &= -A^T p_{M+1}(t) - C^T Q v(t) \\ p_{M+1}(T) &= 0. \end{aligned} \tag{41}$$

Note now that $(PG)_j$ is simply the composite map $G : L_2^e[0, T] \rightarrow L_2^m[0, T]$ and the map

$$y(t) \rightarrow P_j y(t), \quad t \in [t_{j-1}, t_j] \tag{42}$$

which has adjoint

$$v_j(t) \rightarrow \begin{bmatrix} Q^{-1} P_j^T \hat{Q}_j v_j(t), & t \in [\hat{t}_{j-1}, \hat{t}_j] \\ 0, & \text{otherwise} \end{bmatrix} \in L_2^m[0, T]. \tag{43}$$

It follows directly that the adjoint operator P^* is defined by the relation $P^*(v_1, \dots, v_{\hat{M}}) = \hat{w}$, where \hat{w} can be identified almost everywhere with the function

$$\hat{w}(t) = Q^{-1} P_j^T \hat{Q}_j v_j(t), t \in [\hat{t}_{j-1}, \hat{t}_j], \quad t \in [0, T]. \tag{44}$$

Hence, $(PG)^* = G^* P^*$ is defined by a relation of the form $w = (PG)^*(v_1, \dots, v_{\hat{M}})$ as the continuous solution of the costate equation

$$\begin{aligned} w(t) &= R^{-1} B^T p_{M+1}(t) \\ \dot{p}_{M+1}(t) &= -A^T p_{M+1}(t) - C^T P_j^T \hat{Q}_j v_j(t), \quad t \in [\hat{t}_{j-1}, \hat{t}_j] \\ p_{M+1}(T) &= 0. \end{aligned} \tag{45}$$

3) Adjoint operator of G_e : Using these representations, the adjoint operator G_e^* of G_e is the map

$(w_1, \dots, w_M, v_1, \dots, v_{\hat{M}}) \mapsto u$ defined by

$$\begin{aligned} u(t) &= \sum_{j=1}^M (G_j^* w_j)(t) + (G^* P^*(v_1, \dots, v_{\hat{M}}))(t) = \\ &R^{-1} B^T \sum_{j=1}^{M+1} p_j(t) = R^{-1} B^T p(t) \end{aligned} \tag{46}$$

where

$$p(t) = \sum_{j=1}^{M+1} p_j(t) \in \mathbf{R}^n. \tag{47}$$

Using linearity, these can be brought together as

$$\begin{aligned} u(t) &= R^{-1} B^T p(t) \\ \dot{p}(t) &= -A^T p(t) - C^T P_j^T \hat{Q}_j v_j(t), \quad t \in (\hat{t}_{j-1}, \hat{t}_j), \quad 1 \leq j \leq \hat{M} \\ p(T) &= C^T F_M^T Q_M w_M \\ p(t_j-) &= p(t_j+) + C^T F_j^T Q_j w_j, \quad 1 \leq j < M. \end{aligned} \tag{48}$$

This is a familiar equation similar to the costate equations of optimal control. It is computed by backwards integration from the terminal boundary condition at $t = T$ with defined discontinuities (or ‘‘jumps’’) at times $t = t_j, 1 \leq j < M$ and the specified terminal boundary condition. Note that

1) The case created by putting all $P_j = 0$ can be included in the above as it is equivalent to omitting the term in the inner products and norms and redefining the extended output y^e to omit the term $Py(\cdot)$. The output space is then $R^{f_1} \times \dots \times R^{f_M}$ which is finite dimensional of dimension $[f] = \sum_{j=1}^M f_j$. The input space remains infinite dimensional so G_e is not invertible.

2) It is easily seen that the above theory also works with time varying A, B, C, \hat{Q}_j, R simply by replacing these matrices by time varying versions. The details are omitted for brevity.

Substituting the operator form (48) into (17) leads directly to the TPBVP of (27).

References

- [1] S. Arimoto, F. Miyazaki, S. Kawamura. Bettering operation of robots by learning. *Journal of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.
- [2] D. A. Bristow, M. Tharayil, 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.
- [3] H. S. Ahn, Y. Chen, K. L. Moore. Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 37, no. 6, 1099–1121, 2007.
- [4] D. H. Owens, C. T. Freeman, B. Chu. Multivariable norm optimal iterative learning control with auxiliary optimization. *International Journal of Control*, vol. 86, no. 6, pp. 1026–1045, 2013.

- [5] D. H. Owens, C. T. Freeman, T. Van Dinh. Norm optimal iterative learning control with intermediate point weighting: Theory, algorithms, and experimental evaluation. *IEEE Transactions on Control Systems Technology*, vol. 21, no. 3, pp. 999–1007, 2013.
- [6] G. Pipeleers, J. Swevers. A data-driven constrained norm-optimal iterative learning control framework for LTI systems. *IEEE Transactions on Control Systems Technology*, vol. 21, no. 2, pp. 546–551, 2013.
- [7] T. D. Son, H. S. Ahn, K. L. Moore. Iterative learning control in optimal tracking problems with specified data points. *Automatica*, vol. 49, no. 5, pp. 1465–1472, 2013.
- [8] S. H. Zhou, Y. Tan, D. Oetomo, C. T. Freeman, E. Burdet, I. Mareels. Point-to-point learning in human motor systems. In *Proceedings of the American Control Conference*, IEEE, Washington, USA, pp. 5923–5928, 2013.
- [9] C. T. Freeman, T. Exell, K. L. Meadmore, E. Hallewell, A. M. Hughes. Computational models of upper-limb motion during functional reaching tasks for application in FES-based stroke rehabilitation. *Biomedical Engineering*, to be published.
- [10] K. L. Moore, M. Ghosh, Y. Q. Chen. Spatial-based iterative learning control for motion control applications. *Meccanica*, vol. 42, no. 2, pp. 167–175, 2007.
- [11] S. K. Sahoo, S. K. Panda, J. X. Xu. Application of spatial iterative learning control for direct torque control of switched reluctance motor drive. In *Proceedings of IEEE Power Engineering Society General Meeting*, IEEE, Tampa, USA, pp. 1–7, 2007.
- [12] Y. H. Yang, C. L. Chen. Spatial-based adaptive iterative learning control of nonlinear rotary systems with spatially periodic parametric variation. *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 6, pp. 3407–3417, 2011.
- [13] K. Furuta, M. Yamakita. The design of a learning control system for multivariable systems. In *Proceedings of IEEE International Symposium on Intelligent Control*, IEEE, Philadelphia, USA, pp. 371–376, 1987.
- [14] K. Kinoshita, T. Sogo, N. Adachi. Iterative learning control using adjoint systems and stable inversion. *Asian Journal of Control*, vol. 4, no. 1, pp. 60–67, 2002.
- [15] D. H. Owens, J. J. Hatonen, S. Daley. Robust monotone gradient-based discrete-time iterative learning control. *International Journal of Robust and Nonlinear Control*, vol. 19, no. 6, pp. 634–661, 2009.
- [16] N. Amann, D. H. Owens, E. Rogers. Iterative learning control using optimal feedback and feed-forward actions. *International Journal of Control*, vol. 65, no. 2, pp. 277–293, 1996.
- [17] S. Gunnarsson, M. Norrlöf. On the design of ILC algorithms using optimization. *Automatica*, vol. 37, no. 12, pp. 2011–2016, 2001.
- [18] J. H. Lee, K. S. Lee, W. C. Kim. Model-based iterative learning control with a quadratic criterion for time-varying linear systems. *Automatica*, vol. 36, no. 5, pp. 641–657, 2000.
- [19] K. L. Barton, A. G. Alleyne. A norm optimal approach to time-varying ILC with application to a multi-axis robotic testbed. *IEEE Transactions on Control Systems Technology*, vol. 19, no. 1, pp. 166–180, 2011.
- [20] E. Rogers, D. H. Owens, H. Werner, C. T. Freeman, P. L. Lewin, S. Kichhoff, C. Schmidt, G. Lichtenberg. Norm optimal iterative learning control with application to problems in accelerator based free electron lasers and rehabilitation robotics. *European Journal of Control*, vol. 16, no. 5, pp. 497–524, 2010.
- [21] N. Amann, D. H. Owens, E. Rogers. Iterative learning control for discrete-time systems with exponential rate of convergence. *IEE Proceedings of Control Theory and Applications*, vol. 143, no. 2, pp. 217–224, 1996.
- [22] N. Amann, D. H. Owens, E. Rogers. Predictive optimal iterative learning control. *International Journal of Control*, vol. 69, no. 2, pp. 203–226, 1998.
- [23] B. Chu, D. H. Owens. Accelerated norm-optimal iterative learning control algorithms using successive projection. *International Journal of Control*, vol. 82, no. 8, pp. 1469–1484, 2009.
- [24] B. Chu, D. H. Owens. Iterative learning control for constrained linear systems. *International Journal of Control*, vol. 83, no. 7, pp. 1397–1413, 2010.
- [25] D. H. Owens, B. Chu, E. Rogers, C. T. Freeman, and P. L. Lewin. Influence of nonminimum phase zeros on the performance of optimal continuous-time iterative learning control. *IEEE Transactions on Control Systems Technology*, vol. 22, no. 3, pp. 1151–1158, 2014.
- [26] D. H. Owens, B. Chu. Modelling of non-minimum phase effects in discrete-time norm optimal iterative learning control. *International Journal of Control*, vol. 83, no. 10, pp. 2012–2027, 2010.



David H. Owens received the B. Sc. degree in theoretical physics from the Imperial College of Science and Technology, London University, UK in 1969, and the Ph. D. degree in multivariable control of nuclear reactors from Imperial College, UK in 1973. He was the head of the School of Engineering and Computer Science, University of Exeter, UK from 1990 to 1999. He was the

head of the Department of Automatic Control and Systems Engineering from 1999 to 2008 and dean of engineering at Sheffield University, UK from 2002 to 2006. He is currently a visiting professor at School of Electronics and Computer Science, University of Southampton, UK, and a visiting researcher at the Italian Institute of Technology, Italy. He was elected to be a Fellow of the Royal Academy of Engineering in 2008.

His research interests include control systems design, systems modelling, adaptive and robust control, repetitive control, iterative learning systems, nonlinear control systems, and control systems applications.

E-mail: d.h.owens@sheffield.ac.uk



Chris T. Freeman received the B. Eng. degree in electromechanical engineering from University of Southampton, UK in 2000, and the B.Sc. degree in mathematical sciences from the Open University, UK in 2006. In 2004, he received the Ph. D. degree in applied control from the University of Southampton, UK. He is currently an associate professor in electronics and computer science at the University of Southampton, UK. He has published over 200 refereed journal and conference papers. He received an IEEE Control Systems Society Outstanding Paper Award in 2013, and the Best Paper Award of the IEEE International Conference on Rehabilitation Robotics in 2009.

His research interests include the development, application and assessment of intelligent controllers within both the biomed-

ical engineering domain and for application to industrial systems.

E-mail: cf@ecs.soton.ac.uk (Corresponding author)

ORCID iD: 0000-0002-6508-4597



Bing Chu received the B. Eng. degree in automation from Tsinghua University, China in 2004, and the M.Sc. degree in control science and technology from the same institution in 2007. In 2009, he received the Ph. D. degree in automatic control and systems engineering from University of Sheffield, UK. He is currently an assistant professor in electronics and computer science at University of Southampton, UK. He received the Best Paper Award of the United Kingdom Automatic Control Council International Conference on Control in 2012.

His research interests include learning control, analysis and control of networked systems, control design of sustainability development, applied optimisation theory and their practical applications.

His research interests include learning control, analysis and control of networked systems, control design of sustainability development, applied optimisation theory and their practical applications.

E-mail: b.chu@ecs.soton.ac.uk