Error Corrected References for Accelerated Convergence of Low Gain Norm Optimal Iterative Learning Control

David H Owens, Bing Chu

***Abstract*—To reduce the need for high gains (reduced control weighting) for fast convergence in Norm Optimal Iterative Learning Control (NOILC) the paper presents a simple data-driven mechanism for accelerating the convergence of low gain feedback NOILC controllers. The method uses a modification to the reference signal on each NOILC iteration using the measured tracking error from the previous iteration. The basic algorithm is equivalent to a gradient iteration combined with a NOILC iteration. The choice of design parameters is interpreted in terms of the spectrum of the error update operator and the systematic annihilation of spectral components of the error signal. The methods apply widely, including continuous and discrete-time end-point, intermediate point and signal tracking. The effects of parameter choice are revealed using examples. A robustness analysis is presented and illustrated by frequency domain robustness conditions for multi-input, multi-output discrete-time tracking, and robustness conditions for end-point problems for state space systems. Finally, the algorithm is extended to embed a number of gradient iterations within a single NOILC iteration. This makes possible the systematic manipulation of the spectrum, providing additional acceleration capabilities with the theoretical possibility of arbitrary fast convergence.**

***Index Terms*—iterative learning control, performance optimisation.**

I. Introduction

Iterative Learning Control (ILC) is now well established as part of the control scene and many papers have been published [1], [2] plus several texts [3]–[7]. The area of study has the challenges of Control Theory with the added ingredients that (i) the system to be controlled must track an output reference (or demand) signal to an arbitrary high accuracy and (ii) the physical controlled process operates in a repetitive manner. The reference signal could be a function or time series defined on a finite interval, a desired end point, a finite sequence of isolated intermediate points on a time interval or a mixture of of these objectives as in multi-task ILC [7]. Typical applications examples include robotic systems [8], chemical batch processing [9], [10] and stroke rehabilitation [11].

The repetition (or iteration) offers the opportunity to use measured input/output (and other) data from previous iterations to improve on the performance observed in the first attempt to track the reference. Performance improvement is typically expressed in terms of sequentially reducing a tracking error norm and has the ultimate objective of producing much higher accuracy than can be achieved by feedback control alone. For mathematical analysis, the iterations are allowed to be infinite in number. In practice, iterations will be finite in number and the control improvement process will be terminated when the tracking accuracy has been reduced to desired levels.

A number of design paradigms [7], [12]–[31] have been suggested and considerable insight has been obtained into the nature of the control problem and performance limitations. Issues of performance have focused primarily on the convergence of the iterative process. Clearly convergence to a desired solution is part of the requirement but, as each iteration is a physical process in the real world (often involving the organization of machines and operators), the speed of this convergence (as expressed by the number of iterations needed to achieve a desired tracking accuracy) is also important if time and cost are to be within acceptable bounds. This paper focuses on the use of the Norm Optimal Iterative Learning Control (NOILC) algorithm introduced by Amann et al [18] and brought together in the comprehensive text by Owens [7]. This class of algorithms constructs control inputs for each iteration by minimizing a quadratic performance index. Further details are provided in later sections but the essential properties of NOILC for linear processes are that control improvements are guaranteed theoretically from iteration to iteration but that the rate of convergence depends upon the choice of parameters used in the index. For state space systems, the general result is that fast convergence requires the use of high gain state feedback. High gain may be acceptable but, in circumstances such as applications where measurements have some noise component or where modelling errors are significant, high gains may not be acceptable and other mechanisms that use lower gains but retain the convergence speed of higher gains would be preferred. This paper attempts to fill that need.

In this paper, good convergence rates are achieved without high gain control by retaining the NOILC methodology but adding the simple mechanism of changing the reference signal r every iteration in a systematic manner. The original reference r is still the signal to be tracked but, in achieving this objective, it is replaced in NOILC calculations by a sequence of "error corrected" references $\{r_{k+1}\}_{k\geq 0}$,

$$r_{k+1} = r + \beta_{k+1} e_k \tag{1}$$

[1]David H Owens is an Emeritus Professor at the Department of Automatic Control and Systems Engineering, The University of Sheffield, Mappin Street, Sheffield S1 3JD, and was a Principal Scientist with the Department of Automation, Zhengzhou University, Zhengzhou, 450001, P. R. China d.h.owens@shef.ac.uk

[2]Bing Chu is with the School of Electronic and Computer Science, University of Southampton, Southampton, SO17 1BJ, UK b.chu@soton.ac.uk

where r_{k+1} denotes the (virtual) reference signal used on iteration $k+1$, e_k is the observed tracking error on the previous iteration k and β_{k+1} is a scalar design parameter. The NOILC parameters can then be chosen to avoid excessively high gains in an implementation and convergence rates improved by modifications to the reference signal on each iteration. Note that the use of variable reference signals as a design mechanism has not been explored in the literature and is a distinct feature of the approach developed in this paper.

The contribution and organisation of the paper are as follows:

1) The development in the paper summarizes (Section II) the basic ideas of NOILC using the Hilbert space and operator formulation used by Owens [7]. This has the advantage of notational simplicity and generality as many types of linear dynamics and control design problems can be analysed and included. This is followed by a discussion of the related Gradient ILC algorithm.
2) The error corrected algorithm is introduced (Section III) and its properties analysed (Section IV) noting that,
   - if the NOILC algorithm and the Gradient algorithm are applied on alternate iterations, the basic convergence properties of NOILC are still achieved,
   - but that the two iterations steps can be combined to form a new, simplified, iteration that takes the form of NOILC with an error corrected reference signal.

   Not only is the algorithm more rapidly convergent than the use of NOILC alone, the choice of a free parameter in the new algorithm can be approached using the concept of spectral annihilation introduced by Owens in [7]. For state space systems, this offers the opportunity to focus improved convergence rates into selected frequency bands. The important object needed to demonstrate the results is the operator LL_G relating errors on successive iterations. This operator generates a function $f(\eta)$ describing the convergence rates associated with each spectral value η of an operator GG^*. This function reveals the relationship between the choice of parameters and that part of the spectrum that can be annihilated/approximately annihilated. The ideas open up the possibility of more complex updates for the reference signals.
3) Error correction does influence the robustness of NOILC, but a robustness analysis (Section V) characterizes the change in robustness using the concept of "robust monotonic error reduction" (Owens [7]). This analysis reveals that low gain NOILC with error correction can have the same robustness characteristics as the use of higher gain NOILC without error correction.
4) To demonstrate this general design framework, interval tracking and end point control problems for discrete time systems are considered in Section VI with computational aspects presented. Numerical examples are given in Section VII to illustrate algorithm properties.
5) A generalization is considered (in Section VIII) with the aim of achieving further acceleration by combining a finite number of gradient iterations with one standard NOILC iteration to form one NOILC iteration with a modified reference signal. The greater the number of gradient steps used, the more complex will be the off-line computations needed so, ultimately, the generalisation is primarily of theoretical interest. The limits of these possibilities are demonstrated by showing that, in principle, an arbitrary shape for the function f can be achieved, hence achieving an arbitrary shape to the spectral convergence and annihilation properties.

The paper substantially extends our previous conference publication [32] by including i) a more comprehensive analysis of the algorithm's convergence properties, ii) a new section on the robustness analysis, (iii) further generalisations of the design approach, and iv) additional illustrative examples based on applications to interval tracking and end-point control problems.

II. Underlying Optimization Algorithms

A. Models, Signals and Operators

The analysis aims to cover as many applications situations as possible. This is achieved using the original methodology of NOILC proposed in [18] and used more extensively in the text [7]. The plant is assumed to be stable (to permit the collection of data from plant experiments during the iterations). If not, it is assumed that it already includes stabilizing control loops. Its dynamics are assumed to be linear and described by a bounded linear operator G mapping a real Hilbert space $\mathcal{U}$ of input signals u into a real Hilbert space $\mathcal{Y}$ of output signals y. The inner product in $\mathcal{U}$ (resp. $\mathcal{Y}$) is denoted by $\langle u, v\rangle_{\mathcal{U}}$ (resp.$\langle y, w\rangle_{\mathcal{Y}}$). The induced norms in $\mathcal{U}$ (resp. $\mathcal{Y}$) are then given by $\|u\|_{\mathcal{U}} = \langle u, u\rangle_{\mathcal{U}}^{\frac{1}{2}}$ (resp. $\|y\|_{\mathcal{Y}} = \langle y, y\rangle_{\mathcal{Y}}^{\frac{1}{2}}$).

The model equations are represented in the form

$$y = Gu + d, \tag{2}$$

where $d \in \mathcal{Y}$ represents the initial condition and/or repeated disturbances in the system dynamics. As noted in Owens [7], this description covers many situations of practical interest including convolution and state space descriptions of linear continuous time state space models and sampled data and multi-rate sampling systems plus problems where control signals have specific structure exemplified by a requirement that it is continuous and piecewise linear with discontinuities only allowed at specific time instants. In a similar manner to optimal control, the inner products and norms are design variables that can be chosen to reflect physical properties of the system, the control objective and/or the performance of the iterative algorithm to be used.

To illustrate the possibilities of this formulation, a linear, continuous-time, time-invariant, ℓ-input, m-output state space system with input vector $u(t)$ and output vector $y(t)$ can be associated with an input-output convolution relationship

$$y(t) = \int_0^t Ce^{A(t-s)}Bu(s)ds \; + \; Ce^{At}x_0 \; , \quad t \in [0, t_f] \tag{3}$$

which takes the form of (2) where G is the convolution operator defined by the impulse response matrix $H(t) = Ce^{At}B$ and $d(t) = Ce^{At}x_0$. This is the representation used for signal tracking problems. For end-point problems, the same

relationship holds but where t takes the single value t_f. In both cases, the space $\mathcal{U}$ is the Cartesian product of ℓ copies of $L_2[0,t_f]$ with inner product between two inputs u and v

$$\langle u,v\rangle_{\mathcal{U}} = \int_0^{t_f} u^T(t)R(t)v(t)dt \tag{4}$$

where $R(t)$ is symmetric and positive definite for all $t \in [0,t_f]$. For the outputs, $\mathcal{Y}$ could be the product of m copies of $L_2[0,t_f]$ with inner product between outputs y and w

$$\langle y,w\rangle_{\mathcal{Y}} = \int_0^{t_f} y^T(t)Q(t)w(t)dt \tag{5}$$

where $Q(t)$ is symmetric and positive-definite for all $t \in [0,t_f]$. For end-point problems, this would be replaced by

$$\langle y,w\rangle_{\mathcal{Y}} = y^T(t_f)Qw(t_f)\ , \quad Q=Q^T>0\ . \tag{6}$$

In practice [7], $Q(t)$ can be meaningfully positive semi-definite. For example, if tracking of a reference is only required on N_I disjoint, closed sub-intervals $[t_j,t_{j+1}]$ of $[0,t_f]$, then $L_2[0,t_f]$ could be replaced by the Cartesian product of Hilbert spaces $L_2[t_j,t_{j+1}]$. $Q(t)$ should now be positive-definite on these sub-intervals and zero elsewhere.

Finally, other examples including convolution descriptions of discrete time systems, end-point and intermediate point tracking problems can be found in [7] .

B. Iterative Learning Control (ILC)

Iterative Learning Control (ILC) considers a repetitive task whose ultimate objective is to ensure that the output from the system given by equation (2) tracks a given reference signal $r \in \mathcal{Y}$ arbitrarily accurately. Using the formal definition in the text by Owens [7], an ILC algorithm considers an infinite sequence of iterations labelled by the iteration index $k = 0,1,2,3,\cdots$. The iteration with index $k=0$ is the zeroth iteration and is characterized by the choice of an initial control u_0, measurement of the corresponding output y_0 and computation of the error signal $e_0 = r - y_0$. The choice of initial input u_0 is open to the user and could be generated by empirical considerations based on simplicity, experience and/or safety considerations. Alternatively, model-based, off-line calculations based on one of the many feedback or optimal control methodologies could be used. For future iterations, the algorithm uses measurement data from iterations $k' \le k$ plus model-based calculations and feedback, to construct an input u_{k+1} for use on the next iteration $k+1$. The result of this input is the output y_{k+1} and tracking error $e_{k+1} = r - y_{k+1}$.

The design issue for ILC is the construction of u_{k+1} and it is here that different approaches can be taken. For the purposes of this paper, the major requirement of the computation is that,

$$\lim_{k\to\infty} \|e_k\|_{\mathcal{Y}} = 0 \tag{7}$$

which expresses the ideal outcome that the ultimate value of the tracking error is zero. The availability of relevant measurement data is assumed and the convergence of the input sequence $\{u_k\}_{k\ge 0}$ is also desirable. Once achieved, the issues of convergence rates and robustness are important as is an understanding of the nature of the converged input.

C. Norm Optimal Iterative Learning Control (NOILC)

NOILC [7], [18] now has a number of interpretations and extensions, e.g., [33]. In this paper we consider its simplest form. More precisely, given data (u_k, y_k) on the k^{th} iteration, NOILC constructs the input u_{k+1} to be used on iteration $k+1$ by minimizing the quadratic objective function

$$J(u) = \|e\|^2_{\mathcal{Y}} + \epsilon^2\|u-u_k\|^2_{\mathcal{U}} \tag{8}$$

subject to the constraints given by the system dynamics (2). If G is a state space model outlined in Section II-A and $\mathcal{Y}$ and $\mathcal{U}$ are $L_2[0,t_f]$ spaces with inner products defined in that section, this problem is a linear quadratic optimal control problem with a familiar Riccati-style tracking solution.

A theoretical description of the behaviour of the algorithm requires the use of G^*, the adjoint operator of G, i.e. the uniquely defined and bounded linear operator $\mathcal{Y} \to \mathcal{U}$ satisfying the condition that $\langle w, Gu\rangle_{\mathcal{Y}} = \langle G^*w, u\rangle_{\mathcal{U}}$ for all $w \in \mathcal{Y},\ u \in \mathcal{U}$. The calculation of G^* for a number of discrete and continuous time state space problems can be found in [7]. They typically take the form of non-causal state space models with terminal conditions although intermediate point problems also need jump conditions to be included. For example, the adjoint operator to the state space operator defined by (3) with input and output spaces with inner products defined by (4) and (5) has adjoint defined by $v = G^*w$ in the form $v(t) = R^{-1}(t)B^T(t)\psi(t)$ where

$$\dot\psi(t) = -A^T\psi(t) - C^TQ(t)w(t)\ , \quad \psi(t_f) = 0\ . \tag{9}$$

The terminal condition shows that G^* is not causal as the value of $v(t)$ is depends on $w(t')$ for $t' \ge t$. The use of optimal control methodologies makes it possible to construct the minimizing solution for $J(u)$ in a causal form.

Properties of the algorithm include:

1) The input update relationship is
$$u_{k+1} = u_k + \epsilon^{-2}G^*e_{k+1}\ . \tag{10}$$
For state space systems [7], this equation together with the model equations form a two point boundary value problem that can be solved using Riccati methodologies.
2) Operating on (10) with G gives $y_{k+1} = y_k + \epsilon^{-2}GG^*e_{k+1}$. The error sequence therefore satisfies the update relationship $e_{k+1} = Le_k, k \ge 0$ where the operator L,
$$L = (I + \epsilon^{-2}GG^*)^{-1}\ . \tag{11}$$
3) The error signal norm sequence is monotonic in the sense that
$$\|e_{k+1}\|_{\mathcal{Y}} < \|e_k\|_{\mathcal{Y}},\ k \ge 0 \tag{12}$$
until input convergence has been achieved,
4) and the error signal converges to the signal given as the orthogonal projection of e_0 onto the subspace of $\mathcal{Y}$ defined as the kernel of GG^*. If $ker[GG^*] = \{0\}$, then the limit error is precisely zero.
5) The minimum value of $J(u)$ is $J(u_{k+1}) = \langle e_k, Le_k\rangle_{\mathcal{Y}} \le J(u_k) = \|e_k\|^2_{\mathcal{Y}}$.

6) Finally, if the input sequence converges, it converges to a limit u_∞ that minimizes $\|u-u_0\|^2_{\mathcal{U}}$. That is

$$u_\infty \;=\; argmin\{\; \|u-u_0\|^2_{\mathcal{U}} : r = Gu + d \;\} \qquad (13)$$

The parameter ϵ^2 is a measure of the relative weight given to the error and change in control signal. As it decreases, the "controller gain" increases, and any feedback implementation has a "high gain" form. High gain may introduce sensitivity and robustness problems during each iteration so any algorithm that ameliorates this issue will have attractive practical benefit provided that convergence speeds can be retained.

D. Gradient-based Iteration

Gradient methodologies simplify the ILC algorithm structure. Using the approach of Owens in [7], a gradient algorithm uses the update formula,

$$u_{k+1} = u_k + \beta_{k+1} G^* e_k \;, \quad k \geq 0 \;. \qquad (14)$$

Here G^* is the adjoint of G and $\beta_{k+1} > 0$ is a scalar gain parameter (the step length in normal gradient terminology). The error update equation is, $k \geq 0$,

$$e_{k+1} = L_G(\beta_{k+1}) e_k \;,\; L_G(\beta) = (I - \beta G G^*) \;. \qquad (15)$$

The algorithm retains the error monotonicity property if,

$$0 < \beta_{k+1} < 2\|G^*\|^{-2} \;, \quad \forall k \geq 0 \;, \qquad (16)$$

where $\|G^*\|$ is the norm of the operator G^*. The NOILC characterization of the limit also holds true in the case of iteration independent gains [7].

E. A Note on Norm Evaluation

Firstly note that, in all cases, the operator norms satisfy

$$\|G\|^2 = \|G^*\|^2 = \|GG^*\| = \|G^*G\| \qquad (17)$$

which provides alternative ways to characterize the problem. In general, evaluation or estimation of the norm of GG^* (or G^*G) is a useful part of the design process. For example,

1) if GG^* (or G^*G) is a matrix, with eigenvalues $0 \leq \cdots \leq \sigma_2^2 \leq \sigma_1^2$ then its norm is the largest eigenvalue σ_1^2.
2) If the plant is a single-input, single-output (SISO), asymptotically stable discrete state space model characterized by matrices A, B, C with a supervector model [7] defined by a matrix G with u and y in the form of vector time series of length N_0 , then $\mathcal{Y}$ and $\mathcal{U}$ are the Hilbert spaces R^{N_0}. Consider the case when the inner products are

$$\langle y, w\rangle_{\mathcal{Y}} = Qy^T w \;, \quad \langle u, v\rangle_{\mathcal{U}} = Ru^T v \qquad (18)$$

where Q and R are strictly positive scalars, then an upper bound for the norm is obtained from the inequality [7]

$$\|GG^*\| = \|G\|^2 = \|G^*\|^2 \leq \sup_{|z|=1} R^{-1}Q|G(z)|^2 \qquad (19)$$

(equality holding as $N_0 \to \infty$) where $G(z)$ is the transfer function of the system. This characterization provides an important link to more traditional frequency domain approaches to controller design. More general, multi-input multi-output (MIMO) expressions can be found in Owens [7].

III. Error Corrected References and Acceleration of NOILC

A. An Error Corrected Reference Algorithm

In principle, the choice of ILC algorithm can change from iteration to iteration. For example, if iteration one is a gradient iteration and iteration two is a NOILC iteration, the following error update equation and monotonicity properties follow from (11) and (15)

$$e_2 = LL_G(\beta_1)e_0 \;, \quad \|e_2\|_{\mathcal{Y}} \leq \|e_0\|_{\mathcal{Y}} \;. \qquad (20)$$

The corresponding input equations are

$$\begin{array}{ll} u_1 = u_0 + \beta_1 G^* e_0 \;, & (Gradient\ Iteration) \\ u_2 = u_1 + \epsilon^{-2} G^* e_2 \;, & (NOILC\ Iteration) \;. \end{array} \qquad (21)$$

The crucial observation is that the two input equations can be combined to give

$$\begin{aligned} u_2 &= u_0 + \epsilon^{-2} G^* (\epsilon^2 \beta_1 e_0 + e_2) \\ &= u_0 + \epsilon^{-2} G^* (\underbrace{r + \epsilon^2 \beta_1 e_0}_{} - y_2) \end{aligned} \qquad (22)$$

which has the structure of the NOILC update law (10) when $k = 0$ if r is replaced by $r_1 = r + \epsilon^2\beta_1 e_0$ (a change in reference signal) and y_1 and u_1 are replaced by the symbols y_2 and u_2 (a change in labelling). This proves that

Theorem 1 (The Error Corrected Reference): The gradient iteration followed by a NOILC iteration described above can be combined into one NOILC iteration with the reference signal r replaced by the error corrected reference $r_1 = r + \epsilon^2\beta_1 e_0$ and the minimization of

$$J_1(u) = \|r_1 - y\|^2_{\mathcal{Y}} + \epsilon^2 \|u - u_0\|^2_{\mathcal{U}} \;, \qquad (23)$$

which is just $J(u)$ with r replaced by r_1.

Note 1: As the gradient step has been incorporated with an NOILC framework, the input generated by this minimization will be denoted by u_1 (rather than u_2) to synchronize the labelling with that of the error corrected reference signal r_1.

The same argument can be applied to future pairs of gradient plus NOILC iterations and suggests the following algorithm. Note the change in indexing used to reflect the combined gradient and NOILC iteration. Note also that the algorithm allows the use of the open or closed ranges,

$$0 < \beta_{k+1} < 2\|G^*\|^{-2}, \qquad 0 \leq \beta_{k+1} \leq 2\|G^*\|^{-2} \;. \qquad (24)$$

The reason for this is stated in the following, easily proved result:

Proposition 1 (Improvements on NOILC): The self-adjoint operators satisfy the statements,

1) if $\beta = 0$, then $LL_G = L$,
2) $-L \leq LL_G(\beta) \leq L$ if β lies in the closed parameter range $0 \leq \beta \leq 2\|G^*\|^{-2}$ and
3) if the range $\mathcal{R}[G]$ is dense in $\mathcal{Y}$, then $-L < LL_G(\beta) < L$ if β lies in the open parameter range $0 < \beta < 2\|G^*\|^{-2}$.

That is, in the case of an open range of choices for β, the error correction algorithm is guaranteed to improve on the error reduction when compared with standard NOILC. In the closed range, the worst case scenario is that the error norm is no worse that that observed using pure NOILC.

Intuitively, error reduction from the incorporation of a gradient component will require that the open range is used. The closed interval allows $\beta_{k+1} = 0$ which removes the gradient component from the calculation. The benefits of the value $\beta_{k+1} = 2\|G^*\|^{-2}$ will emerge when parameter choice is discussed in terms of the eigenstructure of the operators.

The above leads to the following error corrected algorithm.

Algorithm 1 (AN ERROR CORRECTED NOILC ALGORITHM): Let r be the reference signal. The algorithm chooses $\{\beta_{k+1}\}_{k\geq 0}$ to be an infinite sequence of scalar gains in the closed range defined by (24) and defines the sequence of error corrected reference signals

$$r_{k+1} = r + \epsilon^2 \beta_{k+1} e_k \ , \qquad k \geq 0 \ , \tag{25}$$

where $e_k = r - y_k$ is the actual measured tracking error on iteration k.

The error corrected reference NOILC algorithm is defined by the iterative process described below starting with index $k = 0$:

Step 1 Given the data $\{u_k, y_k, e_k\}$, choose β_{k+1}, compute r_{k+1} and construct the form of the input u_{k+1} by minimizing the performance index

$$J_{k+1}(u) = \|r_{k+1} - y\|^2_{\mathcal{Y}} + \epsilon^2 \|u - u_k\|^2_{\mathcal{U}} \tag{26}$$

subject to the plant dynamics (2). Note that:

a) J_{k+1} is a modified form of the original NOILC performance index (8) with r replaced by r_{k+1}.

b) The solution will typically be a mix of on-line experiment and off-line computations depending on the structure of the system, the controller realization and user choice [7].

Step 2 Use this information to record the new plant input u_{k+1} and observed plant output y_{k+1} and construct the new data set $\{u_{k+1}, y_{k+1}, e_{k+1}\}$.

Step 3 If the error e_{k+1} is satisfactory, terminate the algorithm. Otherwise, continue to the next step.

Step 4 Replace k by $k + 1$ and return to Step 1.

B. Feedback and Feedforward Implementations

As in NOILC [7], [18], the error corrected algorithm can be implemented typically in more than one way. The equation defining the control calculation

$$\begin{aligned} u_{k+1} &= u_k + \epsilon^{-2} G^*(r_{k+1} - y_{k+1}) \\ &= u_k + \epsilon^{-2} G^*(\epsilon^2 \beta_1 e_k + e_{k+1}) \end{aligned} \tag{27}$$

has, at the formal level, a feedback structure which, for dynamical systems, is non-causal and hence cannot be implemented directly. For state space systems it can be transformed into a causal state feedback algorithm using Ricatti methodologies but with r replaced by r_{k+1}.

An alternative approach is to use a model-based, feedforward computation using u_k and the observed error e_k to compute u_{k+1} off-line before application to the system to generate the data for iteration $k + 1$.

The feedforward computations are revealed by substituting the relationship $e_{k+1} = LL_G(\beta_{k+1})$ into (27) to give, after a little manipulation,

$$u_{k+1} = u_k + \epsilon^{-2}(1 + \epsilon^2 \beta_{k+1}) G^* L e_k \ . \tag{28}$$

The important thing to note here is that once Le_k has been computed, G^*Le_k, and hence u_{k+1}, can be evaluated using the model G. Note that Le_k is just the signal $e_k - z_{k+1} = Le_k$ generated by one iteration of a NOILC algorithm to compute $z_{k+1} = Gv_1$ where

$$v_1 = argmin\{\|e_k - z\|^2_{\mathcal{Y}} + \epsilon^2 \|v - v_0\|^2_{\mathcal{U}} : v_0 = 0, z = Gv\}$$

Note that zero initial conditions are required in the evaluation and the signal z is a signal tracking a known "reference" e_k.

IV. Algorithm Properties

A. Error Convergence Properties

Under the conditions defined above, the algorithm generates a sequence of tracking errors $\{e_k\}_{k\geq 0}$ that satisfies the monotonicity property (12) of NOILC. The iterations also satisfy a number of conditions, obtained from the the preceding discussion.

1) The error evolution is described by the relationship

$$e_{k+1} = L_{k+1} e_k, \qquad L_{k+1} = LL_G(\beta_{k+1}) \ . \tag{29}$$

2) Each L_{k+1} is self-adjoint with range $\mathcal{Y}$ and, together with L and GG^*, form a mutually commuting set of operators. They satisfy the inequalities,

$$-I \leq -L \leq L_{k+1} \leq L \leq I \ , \qquad k \geq 0, \tag{30}$$

where the symbol $\leq$ in (30) can be replaced by $<$ if (16) holds true and $ker[GG^*] = \{0\}$ (i.e., if $ker[G^*] = \{0\}$, which implies that the range of G, $\mathcal{R}[G]$, is dense in $\mathcal{Y}$). In particular, as $0 \leq L^2_G(\beta) \leq I$ for all permissible values of β,

$$L^2_{k+1} \leq L^2 \ , \quad \forall \ k \geq 0, \tag{31}$$

so that the product operator

$$\Pi^k_{j=0} L^2_{j+1} \leq L^{2(k+1)} \tag{32}$$

and hence, for all k,

$$\|e_k\|^2 \ \leq \|L^k e_0\|^2 \ . \tag{33}$$

That is, the error norm of the error corrected algorithm, on a given iteration, is always less than or equal to the error, on that iteration, that would be found from using the NOILC methodology starting at e_0.

3) For all $k \geq 0$,

$$\begin{aligned} \mathcal{R}[I - L_{k+1}] = \mathcal{R}[I - L] = \mathcal{R}[GG^*] \ , \\ ker[I - L_{k+1}] = ker[I - L] = ker[GG^*] = ker[G^*] \end{aligned}$$

which, in particular, implies that the orthogonal subspaces $\mathcal{R}[I - L]$ and $ker[I - L]$ are L_{k+1}-invariant for all $k \geq 0$ with $L_{k+1}e = e$ for all $e \in ker[G^*]$.

These properties support the following convergence proof which relies heavily on previous work in Section 5.2.2 in Owens [7] and, in particular, the techniques of Theorem 5.9.

Theorem 2 (Error Convergence): The error corrected NOILC algorithm generates a sequence of error signals that is monotonically decreasing in norm. In addition,

1) If e_0 lies in the closure, $\overline{\mathcal{R}(I-L)} = \overline{\mathcal{R}[GG^*]}$, of the range of $I-L$, then
$$\lim_{k\to\infty} \|e_k\|_{\mathcal{Y}} = 0. \tag{34}$$
2) If e_0 is arbitrary, then the iteration is monotonic with limit
$$\lim_{k\to\infty} e_k = e_\infty = P_{ker[I-L]}e_0 \tag{35}$$
where $P_{ker[I-L]}$ denotes the orthogonal projection of e_0 onto $ker[I-L] = ker[G^*]$ in $\mathcal{Y}$.
3) Finally, if $e_0 \in \mathcal{R}[I-L] = \mathcal{R}[GG^*]$, then the iterates $\{e_k\}_{k\geq 0}$ also satisfy the summability condition
$$\sum_{k\geq 0} \|e_k\|_{\mathcal{Y}}^2 < \infty\ . \tag{36}$$

Proof. The theorem is essentially Theorem 5.9 in the text by Owens [7]. It is proved in a similar way noting that the subspace and invariance properties of each and every one of the $\{L_{k+1}\}_{k\geq 0}$ are identical to those of L. The main change is in the proof of the final statement which is true as, for any integer $p \geq 0$,

$$\begin{aligned}\sum_{k=1}^{p} \|e_k\|^2 &= \sum_{k=1}^{p} \langle \Pi_{j=1}^{k} L_j e_0, \Pi_{j=1}^{k} L_j e_0\rangle \\ &= \sum_{k=1}^{p} \langle e_0, \Pi_{j=1}^{k} L_j^2 e_0\rangle \leq \sum_{k=1}^{p} \langle e_0, L^{2k} e_0\rangle \end{aligned} \tag{37}$$

which gives

$$\sum_{k=1}^{p} \|e_k\|^2 \leq \sum_{k=1}^{p} \|L^j e_0\|^2 \tag{38}$$

which is the error norm summation obtained in NOILC starting at e_0. Using Theorem 5.9 in Owens [7], this is uniformly bounded over values of p and the theorem is proved. ∎

Remark 1: In effect the error corrected algorithm inherits all of the error convergence properties of NOILC. The arbitrary choice of the $\{\beta_{k+1}\}_{k\geq 0}$ within the constraints of (24) permits to use error correction only on selected iterations simply by setting $\beta_{k+1} = 0$ when a NOILC iteration is preferred.

B. Input Convergence Properties

Although the theory proves that the error always converges, the input sequence may not converge to a limit in $\mathcal{U}$ unless $r-d$ lies in the range of G, e.g. for a reference signal that is discontinuous but in $L_2[0,t_f]$, an input to a strictly proper state space system that produces zero error will contain implusive components that cannot be implemented in practice. In this section, suppose that there exists at least one input $u_\infty \in \mathcal{U}$ that satisfies the tracking condition $r = Gu_\infty + d$. If more than one such control exists then they form a linear variety

$$S_u = \{u \in \mathcal{U} : r = Gu + d\} \tag{39}$$

representing the set of all controls that produce a zero tracking error. It can be written as $S_u = \tilde{u} + ker[G]$ for any $\tilde{u} \in S_u$. Under this condition, the input updating equation becomes

$$u_\infty - u_{k+1} = u_\infty - u_k - \epsilon^{-2}G^*(r - y_{k+1} + \epsilon^2\beta_{k+1}(r-y_k))$$

which reduces to the update relationship

$$u_\infty - u_{k+1} = L_0(I - \beta_{k+1}G^*G)(u_\infty - u_k),$$

where

$$L_0 = (I + \epsilon^{-2}G^*G)^{-1}.$$

This equation has the same structure as that for error evolution with G and G^* interchanged and L replaced by L_0.

The following theorem now follows from a simple change of notation and the observation that, if $u_\infty - u \in ker[G]$, it must be true that $u \in S_u$.

Theorem 3 (Input Convergence): The error corrected NOILC algorithm generates inputs with the following properties:

1) If $u_\infty - u_0$ lies in the closure, $\overline{\mathcal{R}(I-L_0)} = \overline{\mathcal{R}[G^*G]}$, of the range of $I-L_0$, then
$$\lim_{k\to\infty} \|u_\infty - u_k\|_{\mathcal{U}} = 0 \quad \& \quad \lim_{k\to\infty} u_k = u_\infty\ . \tag{40}$$
2) If u_0 is arbitrary, then the input iteration has a limit $\hat{u}_\infty \in S_u$ satisfying
$$\lim_{k\to\infty} (u_\infty - u_k) = u_\infty - \hat{u}_\infty = P_{ker[I-L_0]}(u_\infty - u_0)$$
where $P_{ker[I-L_0]}$ is the orthogonal projection of e_0 onto $ker[I-L_0] = ker[G]$ and, as a consequence, u_∞ and $\hat{u}_\infty$ differ only in $ker[G]$.
3) If $u_\infty - u_0 \in \mathcal{R}[I-L_0] = \mathcal{R}[G^*G]$, then the iterates $\{u_\infty - u_k\}_{k\geq 0}$ also satisfy the summability condition
$$\sum_{k\geq 0} \|u_\infty - u_k\|_{\mathcal{Y}}^2 < \infty \quad \& \quad \lim_{k\to\infty} u_k = u_\infty\ . \tag{41}$$

Note 2: As $\mathcal{U}$ is the direct sum of the orthogonal closed subspaces $ker[G]$ and $\overline{\mathcal{R}[G^*G]}$, the theorem covers all possibilities.

The theorem is stated in a parallel form to that for error convergence but the presence of the two inputs u_∞ and $\hat{u}_\infty$ can be removed by noting that u_∞ can be chosen arbitrarily in S_u. Let it be the nearest point to u_0 in S_u. It follows that $\langle u - u_\infty, u_\infty - u_0\rangle = 0\ \forall\ u \in S_u$. This implies that $u_\infty - u_0$ is orthogonal to $ker[G]$ leading to the conclusion that $P_{ker[I-L_0]}(u_\infty - u_0) = 0$ and hence, from the previous theorem, that $\hat{u}_\infty = u_\infty$ which proves the following result,

Theorem 4 (Minimum Norm Property of the Input Limit): The input sequence $\{u_k\}_{k\geq 0}$ converges to a limit u_∞ that retains the minimum input norm NOILC property

$$u_\infty = argmin\{\|u - u_0\|_{\mathcal{U}}^2 : r = Gu + d\} \tag{42}$$

C. Spectral Properties of the Error Convergence

Just as the eigenvalues of a matrix M govern the behaviour of an iteration $z_{k+1} = Mz_k$, $k \geq 0$, the properties of the algorithm presented in this paper are related to spectral properties of the operators $\{L_{k+1}\}_{k\geq 0}$. Using the notation $spec[\Gamma]$ to denote the set of spectral values of an operator

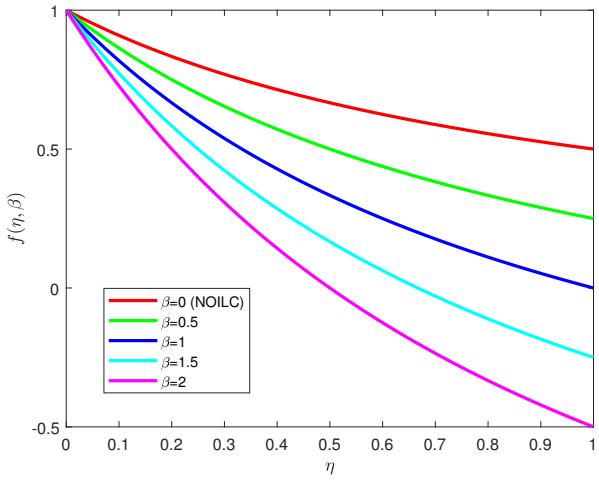


Figure 1. The accelerating effect of β on the spectrum of L_k for Example 1

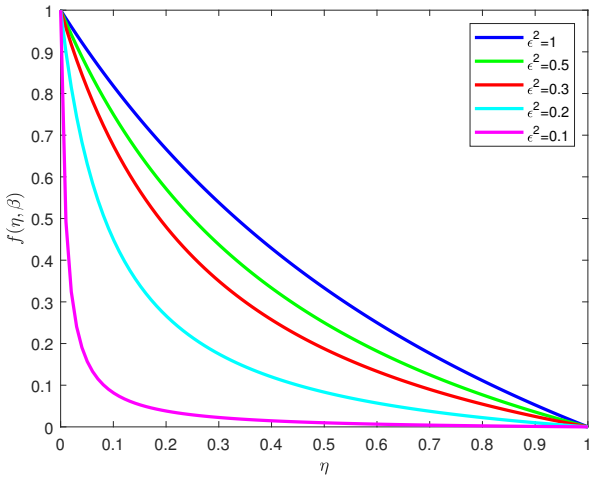


Figure 2. The accelerating effect of ϵ on the spectrum of L_k when $\beta = 1$ for Example 1

$\Gamma : \mathcal{Y} \to \mathcal{Y}$, then the spectrum of L_{k+1} is obtained from the Spectral Mapping Theorem to be exactly

$$spec[L_{k+1}] = \left\{ \frac{1-\beta_{k+1}\eta}{1+\epsilon^{-2}\eta} : \eta \in spec[GG^*] \subset [0, \|GG^*\|] \right\}$$

and hence that the spectrum of L_{k+1} depends on the spectrum of GG^* and lies on the curve

$$f(\eta,\beta) = \frac{1-\beta\eta}{1+\epsilon^{-2}\eta} \ , \quad \beta = \beta_{k+1} \ , \quad \eta \in [0, \|GG^*\|] \ .$$

This function has a simple form dictated by the NOILC parameter ϵ^2 and the choice of β. In particular,

$$f(0,\beta) = 1, \ \forall \ \beta, \quad \& \quad f(\eta,\eta^{-1}) = 0, \ \forall \ \eta \neq 0 \ , \quad \&$$
$$-1 < f(\eta,\beta) < 1, \ \forall \eta \in (0, \|G^*\|^2], 0 \leq \beta < 2\|G^*\|^{-2}.$$

In addition, the special case of the spectral function $f(\eta, 0)$ generates the function for the original NOILC algorithm.

The shape of $f(\eta,\beta)$ is illustrated in the following example and underline the fact that f has modulus less than the NOILC case for all parameters in the ranges of interest, i.e., $|f(\eta,\beta)| < |f(\eta,0)|$.

Example 1: Consider the discrete-time SISO system

$$G(z) = \frac{0.64}{(z-0.2)^2} \ , \quad \sup_{|z|=1} |G(z)| = 1, \tag{43}$$

with zero initial conditions and inner products

$$\langle u, v \rangle_{\mathcal{U}} = \sum_{t=1}^{N} u(t)R(t)v(t) \ , \quad \langle y, w \rangle_{\mathcal{Y}} = \sum_{t=1}^{N} y(t)Q(t)w(t) \ .$$

Take time independent values $Q = R = 1$ and $N = 200$. It follows that $\|G^*\|^2 \approx 1$. Figure 1 shows the plot of $f(\eta,\beta)$ for $\epsilon^2 = 1$ with $\beta = 0 \ , \ 0.5 \ , \ 1 \ , \ 1.5 \ , \ 2$. The first represents NOILC, the third a midpoint of interest and the fifth the largest value of β that could be considered (i.e., $2\|G^*\|^{-2}$). It is clearly seen that for any choice of β, the spectrum of the learning operator is less than that of NOILC and thus faster convergence is expected. For the choice of $\beta = 1$, the effect of weighting ϵ^2 on the spectrum of L_k is shown in Figure 2 where ϵ^2 decreases to take values $1, 0.5, 0.3, 0.2, 0.1$, clearly showing the attenuation effect of control weighting as expected.

To obtain a more detailed insight into the consequences, consider the simplest case when all spectral values are eigenvalues. Let $\{\eta_j\}_{j\geq 0}$ and $\{v_j\}_{j\geq 0}$ be the eigenvalues and orthonormal eigenvectors of GG^*. As GG^* is self-adjoint, all eigenvalues will be real and positive. It follows that L_{k+1} has eigenvectors $\{v_j\}_{j\geq 0}$ and

$$L_{k+1}v_j = f(\eta_j, \beta_{k+1})v_j \ , \quad j \geq 0 \ , \tag{44}$$

and, more generally,

$$L_{k+1}L_k \cdots L_1 v_j = (\Pi_{p=0}^{k} f(\eta_j, \beta_{p+1}))v_j \ , \quad j \geq 0 \ . \tag{45}$$

The range of eigenvalues is given by

$$0 \leq \eta_j \leq \|G^*\|^2 = \|G\|^2 = \|GG^*\| = \|G^*G\| \tag{46}$$

If $\eta_0 = 0$ and the non-zero eigenvalues are ordered $\eta_1 \geq \eta_2 \geq \eta_3 \geq \cdots$, the largest eigenvalue $\eta_1 = \|G^*\|^2$. In particular this means that the parameter range (16) is just

$$0 \leq \beta_{k+1} < 2\eta_1^{-1} \ . \tag{47}$$

In practice, the eigenstructure of GG^* will frequently not be known. Despite this fact, the functional dependence of the spectrum of L_{k+1} on that of GG^* can be used to suggest parameter choices. For example, if the initial error e_0 has the form

$$e_0 = \sum\nolimits_{j\geq 0} a_j v_j \ , \tag{48}$$

then the norm $\|e_0\|^2 = \sum_{j\geq 0} a_j^2 < \infty$. If there are an infinite number of eigenvalues, then the contribution of eigenvectors when j is large is small. If the number of eigenvalues is finite but large in number then it will often be the case that the error will be dominated by a smaller number of eigenvectors. As a consequence, rapid convergence will typically be associated with the rapid suppression of a finite number of eigenvector contributions to e_0.

The convergence of NOILC iterations is revealed by writing

$$e_k = L^k e_0 = \sum_{j\geq 1} a_j v_j \left(\frac{1}{1+\epsilon^{-2}\eta_j}\right)^k = \sum_{j\geq 0} a_j v_j f(\eta_j, 0)^k \ ,$$

which shows that the contribution of each and every eigenvalue/eigenvector reduces geometrically as iterations progress, the smallest eigenvalues being associated with the slowest rate of attenuation. Similarly, for the error corrected algorithm,

$$e_k = \sum\nolimits_{j\geq 0} a_j(\Pi_{p=1}^k f(\eta_j, \beta_p))v_j \ ,$$
$$\|e_k\|^2 = \sum\nolimits_{j\geq 0} a_j^2 \Pi_{p=1}^k f^2(\eta_j, \beta_p) \ ,$$

which, together with $\Pi_{p=1}^k |f(\eta_j, \beta_p)| \leq |f(\eta_j, 0)|^k$, reveals how the contribution of eigenvector components reduce more rapidly than in the NOILC case and how the choice of β_{k+1} influences the reductions. For example, it follows that,

Proposition 2 (Almost Elimination of Eigencomponents):

1) If $\beta_p = \eta_j^{-1}, (j \neq 0)$, then, in the absence of modelling errors, the contribution of v_j to the error response is eliminated for all iterations with index $k \geq p$.
2) For a given choice of β_p, the contribution of the eigenvectors v_j with eigenvalues close to β_p^{-1} will be annihilated or almost annihilated (and/or considerably attenuated) for iterations $k \geq p$.

It also follows that careful selection of the parameters $\beta_k = \eta_k^{-1}$ can, in theory, systematically annihilate the presence of eigenvector components with eigenvalues greater than or equal to $\|G^*\|^2/2$. In practice, of course, the eigenvalues will frequently not be known. However, varying the $\{\beta_{k+1}\}_{k\geq 0}$ over the interval $0 < \beta_{k+1} < 2\|G^*\|^{-2}$ (or, preferably, $\|G^*\|^{-2} < \beta_{k+1} < 2\|G^*\|^{-2}$) will reduce errors faster than the NOILC algorithm and, as f is continuous, almost annihilate all eigenvectors close to the values β_{k+1}^{-1} , $k \geq 0$.

Three Simple Options: based on the availability of an estimate of $\|G^*\|^2$ are as follows,

1) The most cautious choice is to use a value, possibly small, of $\beta_1 \in (0, \|G\|^{-2})$ on all iterations. This will provide some, but limited, acceleration of the algorithm.
2) Choose $\beta_1 \in [\|G\|^{-2}, 2\|G\|^{-2}]$ and use that value for all iterations. For example, choosing $\beta_{k+1} = \|G^*\|^{-2} = \eta_1^{-1}$ for all iterations , immediately eliminates v_1 and provides very rapid reductions of the contributions of the very largest eigenvalues (close to η_1) plus improved reductions in the contributions of smaller eigenvalues when compared with NOILC. This is shown in Figure 3 (for the system of previous examples), and in Figure 4 over two and four iterations. Note that the proposed algorithm, in just two iterations (dotted cyan line in Figure 4), outperforms NOILC over four iterations (red solid line in Figure 4).
3) The third option is to vary β from iteration to iteration. This can improve convergence rates considerably. For example, choosing $\beta_1 = \|G^*\|^{-2}$ and $\beta_2 = 2\|G^*\|^{-2}$ generates an error $e_2 = L_2L_1e_0$ which is associated with the product $f(\eta, 2\|G^*\|^{-2})f(\eta, \|G^*\|^{-2})$ which is shown in the Figure 4 where it is compared with the equivalent NOILC function $f^2(\eta, 0)$ for the case of $\epsilon^2 = \|G^*\|^2$, and the second choice of constant $\beta = 1$. Their performance over four iterations, i.e., the spectra of L^4 and $L_4L_3L_2L_1 = (L_2L_1)^2$, (equivalent to choosing $\beta_3 = \beta_1, \beta_4 = \beta_2$) are also shown. Note the considerable implied improvement in convergence rates for spectral values in the wide interval $[0.2, 1]$. Also note that the choice of varying β outperforms the choice of constant value β, particularly in the range of $[0, 0.5]$.

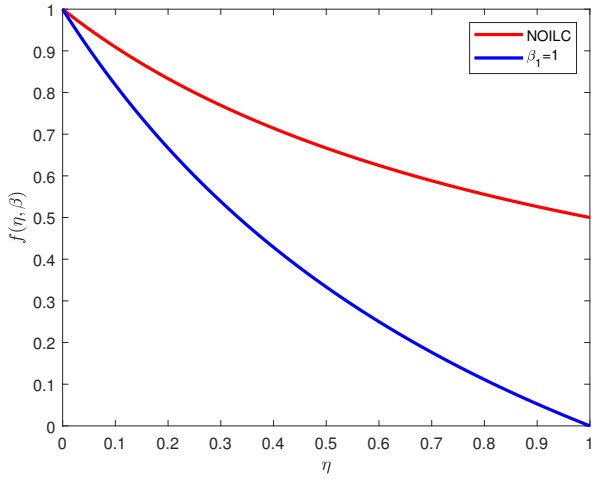


Figure 3. A Comparison of the NOILC spectrum and that of L_1 with $\beta_1 = 1$

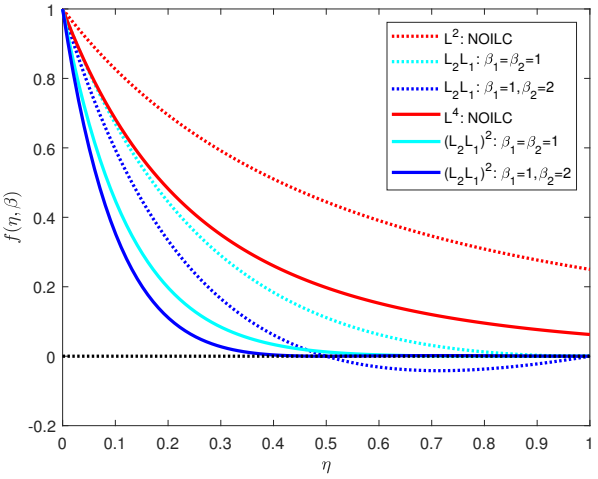


Figure 4. Comparison of the spectrum of $L^2, L^4, L_1^2, L_1^4, L_2L_1, (L_2L_1)^2$ with constant and varying βs

These ideas are explored more fully using simulation examples in the following sections.

V. Algorithm Robustness

A. Choice of Norm and Monotonicity

The previous sections have not considered the situation where the plant model is uncertain. There are many types of uncertainty but, to demonstrate that the error corrected algorithm does possess some robustness properties, this section considers the case where the case of a right multiplicative

modelling error represented by a bounded, linear operator $U : \mathcal{U} \to \mathcal{U}$. The actual plant operator is then GU.

In [7], robustness analyses of several NOILC algorithms are presented based on a modification of the notion that robustness analysis ideally ensures that the algorithm retains the property of monotonicity of the error norm $\|e_k\|_{\mathcal{Y}}$ in the presence of the uncertainty. This problem is difficult but it has been demonstrated [7] that monotonicity of a topologically equivalent norm can often be proved provided that U satisfies some algebraic robustness conditions. The norm $\|\cdot\|_0$ chosen for the error corrected algorithm is defined by,

$$\langle e, w\rangle_0 = \langle e, Lw\rangle_{\mathcal{Y}} \ , \quad \|e\|_0^2 = \langle e, e\rangle_0 \ . \tag{49}$$

The objective of the following analysis is to derive conditions that ensure the monotonic reduction of $\{\|e_k\|_0\}_{k\geq 0}$ as the iterations progress, i.e., to characterize errors U that ensure that

$$\|e_{k+1}\|_0 < \|e_k\|_0, \ k \geq 0 \tag{50}$$

unless $u_{k+1} = u_k$ (when $e_{k+1} = e_k$). Using the inequalities

$$\|e\|_0 \ \leq \ \|e\| \ \leq \ (1 + \epsilon^{-2}\|G\|^2)\|e\|_0 \ , \tag{51}$$

it is seen that the monotonicity of $\|e\|_0$ ensures the boundedness of $\|e\|$ and, in particular, $\|e_k\| \to 0$ if, and only if, $\|e_k\|_0 \to 0$. Also the two norms are almost the same under low gain conditions (represented by $\epsilon^2 \to +\infty$).

B. Robustness of a Feedforward Implementation

The feedforward algorithm is based on the input update (28), and gives $e_{k+1} = e_k - \epsilon^{-2}(1 + \epsilon^2\beta_{k+1})GUG^*Le_k$, so that

$$\|e_{k+1}\|_0^2 = \|e_k\|_0^2 - \epsilon^{-2}(1 + \epsilon^2\beta_{k+1})\Delta E_{k+1} \ ,$$

where

$$\begin{aligned}\Delta E_{k+1} &= \langle G^*Le_k, \Gamma G^*Le_k\rangle_{\mathcal{Y}} \ , \\ \Gamma &= (U + U^*) - (\epsilon^{-2} + \beta_{k+1})U^*G^*LGU \ .\end{aligned} \tag{52}$$

This leads directly to the result that:

Theorem 5 (Algorithm Robustness): The error norm $\|e\|_0$ is always monotonically decreasing on iteration $k + 1$, in the presence of right multiplicative modelling errors U, if

$$(U + U^*) > (\epsilon^{-2} + \beta_{k+1})U^*G^*LGU \ , \tag{53}$$

with respect to the inner product $\langle\cdot,\cdot\rangle_{\mathcal{Y}}$, on the range of G^*. If, in addition, U has a bounded inverse $\hat{U}$ on $\mathcal{U}$, then this condition is equivalent to the requirement that

$$\hat{U} + \hat{U}^* > (\epsilon^{-2} + \beta_{k+1})G^*LG \tag{54}$$

on the subspace $\mathcal{R}[UG^*]$.

Note:

1) As β_{k+1} increases, the set of U satisfying (53) gets smaller indicating a reduction in robustness.
2) For a given value of ϵ^2, the range of values of β_{k+1} that satisfy the conditions are independent of iteration index $k \geq 0$. If at least one value $\beta > 0$ exists, then this range takes the form $\beta \in [0, \beta^*)$ for some $\beta^* > 0$.
3) If $ker[G] = \{0\}$, then $\mathcal{R}[G^*]$ is dense in $\mathcal{U}$. If, in addition, $\mathcal{U}$ is finite dimensional, then $\mathcal{R}[G^*] = \mathcal{U}$. This is the case, for example, for discrete time state space systems and, more generally, any problem where the input signal is defined, linearly, by a finite number of parameters.
4) Replacing the range of G^* (and UG^*) by $\mathcal{U}$ simplifies the checks by removing the dependence on the (unknown) perturbation U. It produces a more conservative result.

The presence of L in the robustness characterization can be removed using either of the two relationships, proved using the Spectral Mapping Theorem,

$$\frac{1}{1 + \epsilon^{-2}\|G\|^2}I \leq L \leq I \ , \quad \& \quad G^*LG \leq \frac{\|G\|^2}{1 + \epsilon^{-2}\|G\|^2}I \ .$$

These produce the following two separate sufficient conditions for (53) to be true,

$$\begin{aligned}&(a) \quad (U + U^*) > \frac{\|G\|^2(\epsilon^{-2} + \beta_{k+1})}{1 + \epsilon^{-2}\|G\|^2}U^*U \ , \\ \textit{or} \quad &(b) \quad (U + U^*) > (\epsilon^{-2} + \beta_{k+1})UG^*GU \ ,\end{aligned} \tag{55}$$

on the subspace $\mathcal{R}[G^*]$.

The first relationship (a) can be written in the form

$$(\zeta^{-1}I - U^*)(\zeta^{-1}I - U) \ < \ \zeta^{-2}I \ , \quad \zeta \ = \ \frac{\|G\|^2(\epsilon^{-2} + \beta_{k+1})}{1 + \epsilon^{-2}\|G\|^2}$$

which can be formalized in the following theorem:

Theorem 6: The Error Corrected Algorithm has the property of robust error norm reduction on all iterations $k \geq 1$ if

$$\|\zeta^{-1}I - U\| < \zeta^{-1} \ , \qquad where \tag{56}$$

$$\zeta \ = \ \frac{\|G\|^2(\epsilon^{-2} + \beta_{max})}{1 + \epsilon^{-2}\|G\|^2} \ , \quad \beta_{max} = \sup_{k\geq 0} \ \beta_{k+1} \ . \tag{57}$$

ζ increases as β_{max} increases. The following result therefore links increases in β_{max} to robustness,

Proposition 3: If U satisfies (56) for a given value ζ_1, it also satisfies the condition for all $\zeta_2 \leq \zeta_1$.

Proof. Write, using norm inequalities,

$$\|\zeta_2^{-1}I - U\| \leq \|\zeta_1^{-1}I - U\| + (\zeta_2^{-1} - \zeta_1^{-1}) < \zeta_2^{-1} \ , \tag{58}$$

as required. ∎

Note that,

1) In later sections, (98) shows that $\zeta \leq 2$.
2) A necessary condition for (56) to be true is that U satisfies the norm bound $\|U\| < 2\zeta^{-1}$.
3) (56) has a similar structure to the robustness conditions found for the inverse ILC algorithm [7].
4) The nominal value of U is $U = I$. Using (55)(a), the requirement that $U = I$ is included in the permissible uncertainty set generates the sufficient requirement that

$$0 \ \leq \ \beta_{k+1} \ < \ 2\|G\|^{-2} + \epsilon^2 \ , \tag{59}$$

which is true for all $\epsilon^2 > 0$ if $\beta_{k+1} \in [0, 2\|G\|^{-2}]$.

A sufficient condition for the existence of parameters satisfying (55)(a) is as follows,

Proposition 4 (Effect of Small Parameter Values): If $U + U^* \geq \delta I$ for some $\delta > 0$, then the error corrected algorithm has the property of monotonically decreasing norm with respect to all small enough values of $\epsilon^{-2} + \beta_{k+1}$, $k \geq 0$.

The condition is a strong form of positive-definiteness of $U+U^*$ in the topology defined by $\langle\cdot,\cdot\rangle_{\mathcal{U}}$ in $\mathcal{U}$ and is linked to the notion of positive real systems.

Finally, if $\beta_{k+1}=0$ the robustness result for NOILC derived in Theorem 9.6 in Section 9.2.2 of [7] is recovered and as $\epsilon^2\to\infty$, $L\to I$, and the characterization reduces to that of the gradient algorithm discussed in Chapter 7 of that reference.

C. State Space Systems: Tracking Problems

Consider now a discrete state space systems (69) with $m\times\ell$ transfer function matrix $G(z)$ using the notation of Section VI-A and replacing $\mathcal{R}[G^*]$ by $\mathcal{U}$. U can then be identified with an $\ell\times\ell$ transfer function matrix $U(z)$. The assumption that $U+U^*\geq\delta I$ for some $\delta>0$ can, if $U(z)$ is asymptotically stable, be replaced by the strictly positive real frequency domain form (Section 4.8 in [7]),

$$\forall\ |z|=1\ ,\quad RU(z)+U^T(z^{-1})R\geq\delta R\ ,\qquad \delta>0\ , \tag{60}$$

the positivity being defined in the Euclidean topology in $\mathcal{C}^\ell$ defined by the inner product $\langle u,v\rangle_{\mathcal{C}^\ell}=\bar{u}^Tv$.

Proposition 5 (Robustness Condition (a)): Suppose that $U(z)$ is asymptotically stable. Consider the robustness condition (55)(a) written as $\|\frac{1}{\zeta}I-U\|<\frac{1}{\zeta}$. This can be written in frequency domain form [7],

$$\sup_{|z|=1} r(\frac{1}{\zeta}I-U(z))(\frac{1}{\zeta}-R^{-1}U^T(z^{-1})R)<\frac{1}{\zeta^2}\ , \tag{61}$$

where $r(\cdot)$ is the spectral radius of the argument, and ζ is defined by (57).

The alternative robustness condition expressed in terms of an asymptotically stable inverse $\hat{U}(z)$ of $U(z)$ is just

$$R\hat{U}(z)+\hat{U}^T(z^{-1})R\ >\ \zeta R,\qquad \forall\ |z|=1\ . \tag{62}$$

For SISO systems, this reduces to the requirement that the Nyquist plot of $\hat{U}(z)$ should lie in the open region of the complex plane to the right of the vertical line $Re[z]=\frac{1}{2}\zeta$.

Now, consider the robustness condition (55) (b) written in the inverse form

$$\hat{U}+\hat{U}^*>(\epsilon^{-2}+\beta_{max})G^*G\quad on\ \ \mathcal{U}\ . \tag{63}$$

This can be converted to a frequency domain form [7], to obtain

Proposition 6 (Robustness Condition (b)): Suppose that both $U(z)$, $\hat{U}(z)$ and $G(z)$ are asymptotically stable. Robustness condition (55)(b) is satisfied in error corrected NOILC tracking problems for discrete state space systems if, $\forall|z|=1$,

$$R\hat{U}(z)+\hat{U}^T(z^{-1})R>(\epsilon^{-2}+\beta_{max})G^T(z^{-1})QG(z)\ ,$$

which is a positivity condition in $\mathcal{C}^\ell$.

Note the following observations:

1) The requirement that both $U(z)$ and $\hat{U}(z)$ are asymptotically stable is equivalent to the requirement that $U(z)$ is both asymptotically stable and minimum-phase.
2) For strictly proper systems, $G(z)$ becomes very small around the high frequency point $z=-1$. The condition therefore allows greater uncertainty in this frequency range before monotonicity is lost.
3) Robustness decreases as the sum $\epsilon^{-2}+\beta_{max}$ increases and stays constant if this sum is constant. Lower gains are introduced if ϵ^2 is increased but the same degree of robustness reclaimed, and some or all of the convergence rate regained, by increasing β. This observation reinforces the assertion that error correction accelerates low gain NOILC with little effect on algorithm robustness.

D. State Space Systems: End-Point Control

Using the notation of Section VI-B, (55)(a) is

$$U+U^*>\zeta U^*U\ ,\qquad on\ the\ subspace\quad \mathcal{R}[G^*]\ , \tag{64}$$

where the parameter ζ is computed using the norm $\|G\|^2=\|GG^*\|$. The evaluation of this quantity is discussed in Section VI-B and is the largest eigenvalue of the $m\times m$ matrix GG^*.

The condition requires verification of the $m\times m$ matrix inequality

$$GUG^*+GU^*G^*>\zeta GU^*UG^*\ . \tag{65}$$

The dependence on U is, however, implicit. This problem is removed by replacing $\mathcal{R}[G^*]$ by $\mathcal{U}$ when the condition

$$\hat{U}+\hat{U}^*>\zeta I\ ,\quad on\quad \mathcal{U}\ , \tag{66}$$

reduces the robustness condition to that of the previous section. It is simply the value of the norm $\|G\|^2$ that changes.

E. Robust Error Convergence

It is known that $\{\|e_k\|_0\}_{k\geq0}$ is monotonic and bounded. In what follows the possibility that $\{\|e_k\|_0\}$ (and hence $\{\|e_k\|_{\mathcal{Y}}\}$ converges to zero is discussed.

Convergence in Norm: Write the error evolution in the form $e_{k+1}=L_U(\beta_{k+1})e_k$ where $L_U(\beta_{k+1})=I-(\epsilon^{-2}+\beta_{k+1})GUG^*L$ and consider the quantity

$$\langle e,L_Ue\rangle_0=\langle e,(L-(\epsilon^{-2}+\beta_{k+1})LGUG^*L)e\rangle_{\mathcal{Y}} \tag{67}$$

Using the robustness condition (53) yields $\langle e,LGUG^*Le\rangle_{\mathcal{Y}}>0,\ \forall e\in\mathcal{Y}$. It follows that $L_U(\beta_{k+1})<I$ in the Hilbert topology in $\mathcal{Y}$ with the inner product $\langle\cdot,\cdot\rangle_0$.

Returning to $\langle e,LGUG^*Le\rangle_{\mathcal{Y}}\ \leq\ \|U\|\langle e,LGG^*Le\rangle_{\mathcal{Y}}$ where $\|U\|$ is the induced operator norm in the Hilbert topology in $\mathcal{U}$ induced by $\langle\cdot,\cdot\rangle_{\mathcal{U}}$. Using the identity $LGG^*=(I-L)\epsilon^2\leq\frac{\|G\|^2}{1+\epsilon^{-2}\|G\|^2}I$ then gives

$$\langle e,L_U(\beta_{k+1})e\rangle_0\geq(1-\|U\|\zeta)\langle e,e\rangle_0 \tag{68}$$

where ζ is defined in (57). It follows that $(-1+\mu)I<L_U(\beta_{k+1})<I$ (in the $\langle\cdot,\cdot\rangle_0$ topology) for any $\mu<2-\zeta\|U\|$. It is easily shown that the existence of such a value $\mu>0$ is guaranteed if the robustness criterion (56) is satisfied.

The relation $(-1+\mu)I<L_U<I,\ \mu>0$ is precisely the condition required to prove convergence of the error, in norm, to zero when the β_{k+1} are iteration independent (Theorem 5.9 in [7]). The iteration dependent result follows in a similar manner. The details are omitted for brevity.

VI. Discrete Time Systems: Interval Tracking & End Point Control Problems

In this section, we discuss the application of the general design framework to two commonly met ILC design problems, namely the interval tracking problem and end point control problem. Consider a linear, time-invariant, ℓ-input, m-output, state dimension n, discrete time state space system with sample interval h on a time interval $[0, t_f]$ with sample points $t = 0, h, 2h, \cdots, Nh = t_f$. Simplifying the notation by assuming that $h = 1$, the model has the classical form,

$$x(t+1) = Ax(t) + Bu(t),\ x(0) = x_0,$$
$$y(t) = Cx(t),\ t = 0, 1, 2, \cdots, N \tag{69}$$

The input time series $\{u(0), u(1), \cdots, u(N-1)\}$ can be regarded as generating the output time series $\{y(1), y(2), \cdots, y(N)\}$.

A. Interval Tracking Problem

The objective is to find an input time series, represented by the column matrix $u = [u^T(0), \cdots, u^T(N-1)]^T$, such that the output $y = [y^T(1), \cdots, y^T(N)]^T$ tracks a reference time series $r = [r^T(1), \cdots, r^T(N)]^T$. The transfer function matrix of the system is $G(z)$ and the associated operator denoted G. The norm $\|G^*\|^2 = \|G\|^2 = \|G^*G\| = \|GG^*\|$ can be estimated using the computations found in Section 4.8.1 in Owens [7]. More precisely, using the Hilbert space $\mathcal{U}$ of inputs with the inner product $\langle u, v\rangle = \sum_{t=0}^{N-1} u^T(t)Rv(t)$ where R is $\ell\times\ell$, symmetric and positive definite and the Hilbert space $\mathcal{Y}$ of outputs with the inner product $\langle y, w\rangle = \sum_{t=1}^{N} y^T(t)Qw(t)$ where Q is $m \times m$, symmetric and positive definite, the operator norm is bounded [7] by

$$\|G\|^2 \leq \sup_{|z|=1} r\left(R^{-1}G^T(z^{-1})QG(z)\right)\ , \tag{70}$$

where $r(\cdot)$ is the spectral radius of the argument. The accuracy of the upper bound increases as the length of the time interval Nh increases, becoming exact when $N \to \infty$.

Given a starting input time series u_0 and the error e_0, iterations are based, for all $k \geq 0$, on tracking the error corrected reference $r(t) + \epsilon^2\beta_{k+1}e_k(t)\ ,\ t \geq 0$ by using optimal control methods to minimize the objective function

$$J = \sum_{t=1}^{N}(\ (r_{k+1}(t) - y(t))^T Q(r_{k+1}(t) - y(t))$$
$$+\epsilon^2(u(t-1) - u_k(t-1))^T Q(u(t-1) - u_k(t-1))\ ) \tag{71}$$

The choice of the parameters $\{\beta_{k+1}\}_{k\geq 0}$ can be based on the ideas outlined in the section on endpoint control but, for tracking problems of the type discussed here, there is a useful frequency domain interpretation. The details can be found in Section 7.2.3 in the text by Owens [7] but, in essence, a choice of β will almost annihilate eigencomponents of GG^* at all frequencies where the gain of the eigenvalues of $G(z)R^{-1}G^T(z^{-1})Q$ take the value β^{-1}. The details are omitted here for brevity but the simplest case of $SISO$ systems can be summarised as follows: if z_j are the distinct N^{th} complex roots of unity then $W(z_j) = [1, z_j, \cdots, z_j^{N-1}]^T$, $1 \leq j \leq N$, are a (complex) basis for $\mathcal{Y}$ representing "sinusoidal " variation. The useful relationship here is as follows,

$$GG^*W(z_j) \simeq R^{-1}Q|G(z_j)|^2W(z_j)\ , \quad 1 \leq j \leq N\ , \tag{72}$$

the approximation being increasingly accurate as N increases. That is, the squared modulus of $G(z)$ can be regarded as an "approximate eigenvalue" and the oscillatory time series $W(z_j)$ as an approximate eigenvector. This suggests that choosing $\beta = RQ^{-1}|G(z)|^{-2}$ will approximately annihilate the frequency component of the error at the frequency z. The is illustrated in the examples of the next section.

B. End Point Control Problem

Consider the case of end point control [7], [34], [35] where it is desired that the control sequence is chosen to reach a desired end point $y(N) = y_f$. In ILC terms, the aim is to find the input time series column vector $u = [u^T(0), \cdots, u^T(N-1)]^T$ that takes the output $y = y(N)$ to the final value $r = y_f$. The error on the k^{th} iteration will simply be

$$e_k = r - y_k = y_f - y_k(N)\ . \tag{73}$$

The dynamics has the alternative form $y = Gu + d$ where

$$G = [CA^{N-1}B, \cdots, CAB, CB], \quad d = CA^N x_0.$$

In terms of the notation of previous sections, use the inner product $\langle y, w\rangle = y^TQw$ in the output Hilbert space $\mathcal{R}^m$ of $m \times 1$ columns of real numbers. The $m \times m$ matrix Q is symmetric and positive definite. The Hilbert space of inputs will be taken to be of the form defined by the inner product $\langle u, v\rangle = u^TRv$ in the Hilbert space $\mathcal{R}^{\ell(N-1)}$. Typically R is block diagonal of the form $R = blkdiag[R_1, R_2, \cdots, R_N]$, where each R_j is symmetric and positive definite. From the text by Owens [7], the adjoint operator

$$G^* = R^{-1}G^TQ \qquad \Rightarrow \qquad GG^* = GR^{-1}G^TQ \tag{74}$$

which is an $m \times m$ matrix with m positive, real eigenvalues $\eta_1 \geq \eta_2 \geq \cdots, \eta_m \geq 0$ with $\eta_m > 0$ if and only if G has full rank. This matrix (and its eigenstructure) can be estimated using a model and the quantity $\|GG^*\| = \|G^*G\| = \|G\|^2 = \|G^*\|^2$ set equal to the largest eigenvalue. The case of single-input, single output (SISO) systems with $m = \ell = 1$ and Q and R multiples of the identity, gives

$$\|GG^*\| = \|GR^{-1}G^TQ\| = R^{-1}Q\|H\|^2\ , \tag{75}$$

where $\|H\|^2 = \sum_{t=1}^{N}(CA^{t-1}B)^2$ is the squared Euclidean norm of the impulse response sequence for the system truncated to the interval of interest.

The operators and norms associated with error evolution for NOILC and the error corrected algorithm are $m \times m$ matrices,

$$L = (I + \epsilon^{-2}GR^{-1}G^TQ)^{-1},\ L_G(\beta) = I - \beta GR^{-1}G^TQ\ ,$$
$$\|G^*\|^2 = \eta_1,\ 0 \leq \beta \leq 2\eta_1^{-1}\ .$$

The range of GG^* is the span of eigenvectors of $GR^{-1}G^TQ$ associated with non-zero eigenvalues whilst the kernel of GG^* is the span of eigenvectors having zero eigenvalues. In order

that all possible output final values y_f can be reached, all eigenvalues must be non-zero.

Starting with an initial control sequence, u_0, the problem to be solved for the $(k+1)^{th}$ iteration is, given the error from the k^{th} iteration and a choice of β_{k+1}, to construct the error $e_k = y_f - y_k$ and the corrected reference $r_{k+1} = y_f + \epsilon^2\beta_{k+1}e_k$ and then to find the input time series vector u_{k+1} minimizing

$$J = (r_{k+1} - y(N))^T Q(r_{k+1} - y(N)) + \epsilon^2 \sum_{t=0}^{N-1} (u(t) - u_k(t))^T R_{t+1}(u(t) - u_k(t)) \ . \quad (76)$$

This is a well defined optimal control problem that can be computed off-line (a feedforward solution) or solved on-line (a feedback solution) using Riccati matrix methods.

For end point problems with only non-zero eigenvalues, the first thing to note is that the choice of the parameters $\beta_p = \eta_p^{-1}$, $p = 1, \cdots, m$ will, in theory reduce the end-point error to zero on the m^{th} iteration $k = m$. More generally, choosing β_p, sequentially, to be equal to the distinct eigenvalues will reduce the end-point error to zero in $\leq m$ iterations. In the process, the algorithm finds the control that transfers the system from its initial state to the desired final output, using a control time series that minimizes

$$\|u - u_0\|^2 = \sum_{t=0}^{N-1} (u(t) - u_0(t))^T R_{t+1}(u(t) - u_0(t)). \quad (77)$$

The $SISO$ case will achieve the objective in one iteration using the error corrected reference $r_1 = y_f + \epsilon^2 R^{-1}Q\|H\|^2 e_0$. More generally, the eigenvalues are not known but, if the norm $\|GG^*\|$ can be estimated, an approximation to the desired range of parameters β_{k+1} will be available.

Finally, note that,

1) u_0 will influence the form of the converged input. It is natural to use one of a desirable or convenient form even if it fails to meet the end point condition. For example, u_0 could be generated by a single NOILC iteration attempting to track a nominal reference r_{nom} time series starting at Cx_0 and terminating at $y(N) = y_f$. The nominal trajectory could be chosen to represent the sort of motion that fits the characteristics of the application.
2) The intermediate point control problem [7], [24], [26] also lead to relatively low order matrix descriptions of L and GG^* of dimension $\ll N$ so similar comments can be made for this case but are omitted for brevity.

VII. Illustrative Computational Examples

In this section, we provide several numerical examples to illustrate the performance of the design algorithms.

A. Interval Tracking Problems

The following examples are designed to verify the validity of the theory and its interpretation. In all cases, $\epsilon^2 = 1$.

Example 2: **Annihilating a Single Frequency/Spectral Value.** As noted above, in [7], it is shown that, for a discrete SISO state space system (A, B, C) operating on an interval $t = 0, 1, 2, \cdots, N$ with transfer function $G(z)$ and a suitable choice of sinusoidal vector $\alpha(z) = \{1, z, z^2, \cdots, z^N\}$, the relation

$$\left(GG^* - R^{-1}Q|G(z)|^2 I\right)\alpha(z) \approx 0 \ , \quad (78)$$

with good accuracy if control is on a long enough interval. The interpretation of this is that $\alpha(z)$ is an approximate eigenvector of GG^* with approximate eigenvalue $R^{-1}Q|G(z)|^2$. This suggests that the choice of

$$\beta_1 = RQ^{-1}|G(z)|^{-2} \quad (79)$$

will almost annihilate an initial error $\alpha(z)$ in one iteration. For example, considering the linear SISO discrete system with transfer function

$$G(z) = \frac{0.7}{z - 0.7} \ , \quad \sup_{|z|=1} |G(z)| = 2.333 \ . \quad (80)$$

Zero initial conditions are assumed and a reference signal defined by the oscillating time series

$$r = \{r(t)\}_{0 \leq t \leq N} \ , \quad r(t) = \frac{1}{c} sin(\frac{\pi}{2}t) = \frac{1}{c} Im\left[i^t\right] \ , \quad (81)$$

with a frequency $z = i$, trial length $N = 2000$, and a constant $c = 31.6228$ normalising the reference norm (to be unity). The weights in the performance index are chosen as $Q = R = 1$.

Approximate annihilation of the single frequency reference is approached by using (79) to choose $\beta_{k+1} = 3 \approx RQ^{-1}|G(i)|^{-2}$. The algorithm is initiated with $u_0 = 0$. The first iteration gives the error norm values

$$\|e_0\| = \|r\| = 1 \ , \ \|e_1\| = 0.0222. \quad (82)$$

The substantial reduction in error provides a simple illustration of the approximate annihilation properties of the new algorithm. Note the gain used does not satisfy (16) and is not necessarily an acceptable choice for further iterations.

Example 3: **Annihilating a Frequency/Spectral Band.** Reference signals, in most cases, contain a wide band of frequencies and plant dynamics are more complex. A simple illustration of possible approaches to such cases is the choice

$$G(z) = \frac{0.64}{(z - 0.2)^2} \quad (83)$$

with zero initial conditions and a reference signal $r(t) = t, \ t = 1, 2, \cdots, N$.

Again take $Q = R = 1$ and $N = 200$. Then (19) suggests that $\|G^*\|^2 \approx 1$. Taking the initial control $u_0 = 0$, the results obtained for NOILC and two error corrected approaches are shown in the plot of the evolution of the error norm in Fig. 5. The two approaches are:

1) Approach One: Take $\beta_{k+1} = \beta$ to be iteration independent and let $\beta = 1$ to approximately annihilate the contribution of the eigenvectors corresponding to the largest eigenvalues of GG^*. Alternatively, (78), the choice will approximately annihilate the DC and part of the low frequency bandwidth content of the error. Fig. 5 confirms the initial and long term improvements in rate of convergence compared with NOILC. Given the simplicity of the error correction mechanism, its benefits are clear.

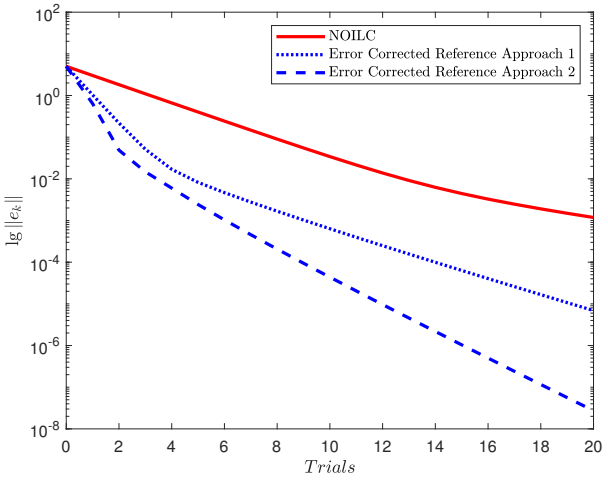


Figure 5. Logarithmic error norm convergence for NOILC and error corrected algorithms showing order of magnitude increases in convergence rates for Example 4

2) Approach Two: By letting the gains vary from iteration to iteration, different frequency contributions can be approximately annihilated. The constraint of (24) is taken to ensure that overall error reduction is guaranteed as the iterations progress. This means that we must focus on the frequencies where

$$0.5 \leq |G(z)|^2 \leq 1 \ . \tag{84}$$

The values of these frequencies are not required for the choice of β_{k+1}. All that is needed is to choose $\beta_{k+1} \in [1, 2]$. To illustrate the possibilities, the choices are made to 'cover' this interval, for example,

$$\beta_1 = 1.0, \beta_2 = 1.2, \beta_3 = 1.4, \beta_4 = 1.6, \beta_5 = 1.8 \tag{85}$$

and, for simplicity, $\beta_{k+1} = 2.0$ thereafter. As the gain of $G(z)$ decreases with frequency, the motivation for this choice is to sequentially and approximately annihilate the frequency content of the error in the range represented by (84), beginning with the lowest frequency.

The algorithm behaviour is shown in Fig. 5 . Although NOILC and Approach One produce monotonic reductions in norm, varying β_{k+1} further accelerates the algorithm producing even more rapid reductions in error norm.

The two approaches used for gain selection show different behaviours. For the first iteration, the norm reduction is the same (as the gain value is the same). For Approach One, error norm reductions are two orders of magnitude better than NOILC over twenty iterations. For further iterations, varying the gains in Approach Two adds to the acceleration by a further two orders of magnitude as it is more effective in eliminating frequency content in the range (84).

B. End Point Control

The following example illustrates the the error corrected algorithm for end point control. Note the weighting chosen is $\epsilon^2 = 1$.

Example 4: Consider the state space model (69) with

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.001 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.001 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.001 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.001 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.001 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.001 & 1 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.001 & 5 \times 10^{-7} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.001 & 5 \times 10^{-7} & 0 & 0 \end{bmatrix}^T$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

with $x_0 = 0$ and a sampling time of $h = 0.001$s. This is a model used in functional electrical stimulation based stroke rehabilitation after applying a feedback linearisation controller (see [36] for more detail). The input to this model is the electrical stimulation levels applied to two pairs of arm muscles, and the two outputs are angle positions of two arm joints. The design objective is to compute the input such that the output reaches the following desired position at $t_f = 6$s (or equivalently at sample number $N = 6000$)

$$y_f = \begin{bmatrix} -0.3 \\ -0.6 \end{bmatrix}.$$

With the choice of $R = I, Q = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$ it is easy to compute

$$GG^* = \begin{bmatrix} 2.2217 & 0 \\ 0 & 1.1109 \end{bmatrix},$$

whose eigenvalues are $\eta_1 = 2.2217, \eta_2 = 1.1109$. From the above discussion, if we choose $\beta_1 = \eta_1^{-1} = 0.4501, \beta_2 = \eta_2^{-1} = 0.9002$, the tracking error will be eliminated in 2 iterations. To show this, if we choose $u_0 = 0$, the tracking error can be computed as follows:

$$e_0 = \begin{bmatrix} -0.3 \\ -0.6 \end{bmatrix}, \ e_1 = \begin{bmatrix} 0 \\ -0.1421 \end{bmatrix}, \ e_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

confirming convergence to zero error in two iterations. This is true for all choices of u_0 and almost all R and Q. Exceptions include, for example, the choice of $R = I$ and Q as a multiple of the identity matrix when convergence is achieved in *one* iteration!

The examples provide evidence that the acceleration produced by the simple, easily implemented, error correction mechanism can, in the absence of modelling errors, be substantial.

VIII. Spectral Manipulation: A Generalization

A. A Modified Error Corrected Reference Algorithm

It is natural to consider the possibility of undertaking m_G off-line gradient steps between each on-line NOILC iteration and ask whether or not the process can be replaced by a single error corrected NOILC iteration.

On the $(k+1)^{th}$ iteration, the error resulting from the gradient calculations has the form $(I - GG^*\Phi(GG^*))e_k$ where

$\Phi(\lambda)$ is a polynomial of degree $m_G - 1$ with $\Phi(0) > 0$. More precisely,

$$\Phi(\lambda) = \lambda^{-1}\left(1 - \prod_{j=1}^{m_G}(1 - \beta_{k+1}^{(j)}\lambda)\right) \ , \tag{86}$$

where $\{\beta_{k+1}^{(j)}\}$ are the gradient gains used in the m_G gradient steps. This demonstrates that the gradient steps are equivalent to a single gradient update with

$$u_k \to u_k + G^*\Phi(GG^*)e_k \ . \tag{87}$$

Following the gradient steps by a NOILC calculation, suggests an input update equation of the form

$$u_{k+1} = u_k + \epsilon^{-2}G^*(\epsilon^2\Phi(GG^*)e_k + e_{k+1}) \tag{88}$$

which becomes an error corrected reference algorithm with

$$r_{k+1} = r + \epsilon^2\Phi(GG^*)e_k \ , \qquad k \geq 0 \ . \tag{89}$$

The calculation of r_{k+1} is based on model-based computation of the terms such as $(GG^*)^p e_k$, $1 \leq p \leq m_G - 1$, typically using simulation methods. The resultant relationship between successive errors takes the form $e_{k+1} = LL_G e_k$ with L_G defined by

$$L_G = \prod_{j=1}^{m_G}(I - \beta_{k+1}^{(j)}GG^*) = I - GG^*\Phi(GG^*) \ . \tag{90}$$

B. Properties of the Algorithm

If each of the gradient steps reduces the error norm, it follows that $-I \leq L_G \leq I$, from which

$$-L \leq LL_G \leq L \ . \tag{91}$$

Hence, using techniques to those used earlier, the error norm sequence reduces faster than the standard NOILC algorithm and has identical convergence properties. The details are omitted for brevity.

The nature of the convergence depends on the choice of gradient parameters and the spectrum of LL_G,

$$spec[LL_G] = \{f(\eta) : \eta \in spec[GG^*]\} \subset (-1, 1] \ .$$

Define, for $0 \leq \eta \leq \|G\|^2$, the function f,

$$\frac{(-1)}{1+\epsilon^{-2}\eta} \leq f(\eta) = \frac{1-\eta\Phi(\eta)}{1+\epsilon^{-2}\eta} \leq \frac{1}{1+\epsilon^{-2}\eta} \tag{92}$$

The form of f describes the rate of convergence of the component of the error associated with the spectral value η.

Example 5: Consider the same system (and the settings) as in Example 1. If $m_G = 2$ and $L_G = (I-\alpha_1 GG^*)(I-\alpha_2 GG^*)$ with $\alpha_j \in [0, 2\|G\|^{-2}]$, $j = 1, 2$, then $\Phi(\lambda) = (\alpha_1 + \alpha_2) - \alpha_1\alpha_2\lambda$. Choosing $\alpha_1 = 1$ and $\alpha_2 = 2$ produces an $f(\eta)$ of the form illustrated in the Figure 6 and compared with the form for a single standard NOILC iteration. Note that, in principle, the improvement in convergence speed is substantial.

In conclusion the error corrected algorithms has the potential to produce very rapid convergence which will reduce the on-line gains needed in the NOILC phase by allowing an increase in the value of ϵ^2. In effect, error correction allows fast convergence without high gain feedback.

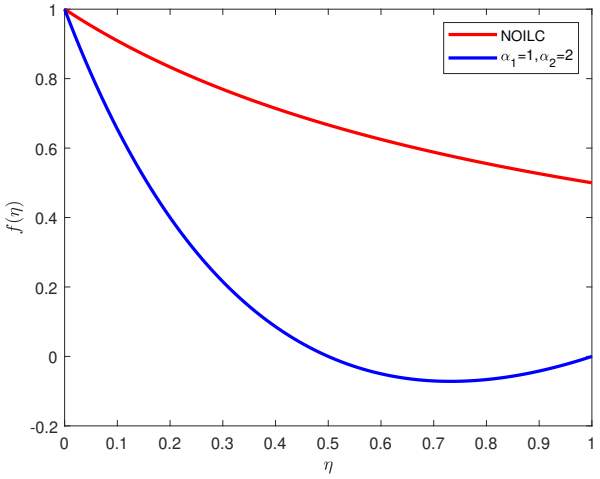


Figure 6. The effect of two gradient updates on the spectrum spectral function $f(\eta)$ when $\alpha_1 = 1, \alpha_2 = 2$ for Example 5

C. Robustness Properties

The robustness of the algorithm in the presence of a right multiplicative perturbation U is affected by the more general form of $\Phi_{k+1}(\eta)$. The feedforward algorithm is described by

$$\begin{aligned} u_{k+1} &= u_k + \epsilon^{-2}G^*(\epsilon^2\Phi_{k+1}(GG^*) + LL_G)e_k \ , \\ e_{k+1} &= e_k - \epsilon^{-2}GUG^*(\epsilon^2\Phi_{k+1}(GG^*) + LL_G)e_k \\ &= e_k - GUG^*(\Phi_{k+1}(GG^*) + \epsilon^{-2}I)Le_k \end{aligned} \tag{93}$$

Robustness is expressed in terms of the monotonic reduction of an appropriate norm. The inner product chosen is

$$\langle e, w\rangle_0 = \langle e, (\Phi(GG^*) + \epsilon^{-2}I)Lw\rangle_{\mathcal{Y}} \tag{94}$$

and the induced norm $\|e\|_0^2 = \langle e, e\rangle_0$ which is a norm as the gradient property requires that $GG^*\Phi(GG^*) \geq 0$ and hence that $\Phi(GG^*) \geq 0$. With the plant dynamics GU this yields, after a little manipulation,

$$\|e_{k+1}\|_0^2 \ = \ \|e_k\|_0^2 - \langle v, \Gamma v\rangle_{\mathcal{Y}} \ , \tag{95}$$

where

$$\begin{aligned} v &= G^*(\Phi_{k+1}(GG^*) + \epsilon^{-2}I)Le_k, \\ \Gamma &= U + U^* - U^*GL(\Phi(GG^*) + \epsilon^{-2}I)GU. \end{aligned}$$

As a consequence, the plant algorithm has the property of robust monotonicity if $\Gamma > 0$ i.e.

$$U + U^* > U^*GL(\Phi(GG^*) + \epsilon^{-2}I)GU \tag{96}$$

on $\mathcal{R}[G^*]$ with respect to the inner product $\langle\cdot,\cdot\rangle_{\mathcal{Y}}$.

Using the property,

$$G^*L(\Phi(GG^*) + \epsilon^{-2}I)G \leq \zeta I \quad where \ , \tag{97}$$

$$\begin{aligned} 0 \leq \zeta \ &= \ \sup\{\ \tfrac{\eta(\Phi(\eta)+\epsilon^{-2})}{1+\epsilon^{-2}\eta} : \eta \in [0, \|G\|^2]\} \\ &= \ \sup\{\ 1 - f(\eta) : \eta \in [0, \|G\|^2]\} \\ &\leq \ \sup\{\ \tfrac{2+\epsilon^{-2}\eta}{1+\epsilon^{-2}\eta} : \eta \in [0, \|G\|^2]\} \ = \ 2 \ , \end{aligned} \tag{98}$$

yields the familiar sufficient condition,

$$U + U^* \ > \ \zeta UU^* \ , \tag{99}$$

which is identical in form to that seen in Section V.

Note 3: Finally, the predicted degree of robustness is uncertain as the robustness condition is sufficient but not necessary. The fact that $\zeta \leq 2$ does indicate, however, that there will be a degree of robustness for all permissible parameter choices.

D. The Theoretical Possibilities when choosing $m_G > 1$

The form of the function $f(\eta)$ links the spectrum of the algorithm evolution operator LL_G to convergence properties. Intuitively, as m_G increases, the shape of f and hence convergence characteristics may change significantly. This short section investigates just how much control the choice of Φ can have on the form of f. In what follows, Φ is regarded as a design "parameter" rather than a consequence of gradient computations. It addresses the question of shaping the function $f(\eta)$ under the constraint that the algorithm improves on standard NOILC. More precisely, given a desired, continuously differentiable function $f_d(\eta)$ that improves on the NOILC function $f_N = (1+\epsilon^{-2}\eta)^{-1}$ in the sense that

$$(i) \quad -f_N(\eta) < f_d(\eta) < f_N(\eta)\ , \quad \eta \in (0, \|G\|^2]\ ,$$
$$(ii) \quad f_d(0) = 1\ , \qquad (iii) \quad \frac{df_d}{d\eta}|_{\eta=0} < \frac{df_N}{d\eta}|_{\eta=0}\ , \quad (100)$$

is it possible to choose a polynomial $\Phi(\eta)$ such that f satisfies these constraints and also approximates f_d to an arbitrary accuracy on $[0, \|G\|^2]$? The norm used to represent the accuracy is that used for continuous functions, $\|f - f_d\|_c = \sup\{\ |f(\eta) - f_d(\eta)|\ :\ \eta \in [0, \|G\|^2]\ \}$.

Theorem 7 (Spectral Shaping): With the above notation, there exists a polynomial $\Phi(\eta)$ such that the spectral function $f(\eta)$ associated with the operator $L(I - GG^*\Phi(GG^*))$ is arbitrarily close to $f_d(\eta)$ with respect to the $\|\cdot\|_c$ norm and satisfies the conditions (100).

Note 4: There is no requirement that $I - \eta\Phi(\eta)$ has roots that lie only in $[-1, 1]$ for $\eta \in [0, 2\|G\|^{-2}]$, or even that its roots are real, so the choice may not have a gradient interpretation.

Proof. $\Psi_d = f_N^{-1} f_d$ is continuously differentiable with a derivative that can, using the Weierstrass Theorem, be arbitrarily accurately approximated by a polynomial $\Psi_d(\eta)$ on $[0, \|G\|^2]$. Construct Φ from $\Psi(\eta) = 1 + \int_0^\eta \Psi_d(\sigma)d\sigma = 1 - \eta\Phi(\eta)$ yields the result for all sufficiently accurate approximations. ∎

In effect, the error corrected algorithm provides a large degree of control over the form of the spectral function $f(\eta)$. Previous sections have seen the potential benefits of this using $m_G = 1, 2$ but it is an open question as to whether, in practice, the flexibilities in the use of even higher values merits the additional computational load.

IX. Conclusions

The paper has addressed the practical issue that NOILC algorithms can achieve rapid convergence rates only by using "high gain" feedback solutions (represented by a small control weight factor ϵ^2). It has been shown that a combination of NOILC and Gradient ILC into one iterative step yields a highly attractive approach in the form of a simple and easy to implement modification to the reference signal from iteration to iteration using the previously observed tracking error. The approach chosen is that used by Owens [7] which shows that the ideas are quite general applying to a range of NOILC applications including tracking and end-point problems for continuous and discrete state space systems, the more general "multi-task" problems introduced by Owens in [7], to static/algebraic problems and to delay systems and similar complications. In all cases the modified algorithm inherits the properties of NOILC including error convergence and input convergence to a minimum norm solution.

Two additional parameters are used. In the simple form presented, it is useful to have an estimate of the squared norm $\|G\|^2$ of the plant operator. Fortunately, such norms can often be estimated using a plant model. Another parameter, $\beta_{k+1} > 0$, is chosen on iteration $k+1$. It is generated from gradient algorithm considerations but choosing values can be, more usefully, related to the idea of annihilating the contribution of selected parts of the spectrum of the operator GG^* to the tracking error. The effect of the error correction can be assessed using the graphical form of a function $f(\eta)$ representing the attenuation of the contribution of a spectral value η. For state space systems, the choice can be linked to the systematic removal of frequency components from the error. Simple examples verify the ideas and demonstrate the potential benefits of careful parameter selection.

A robustness analysis based on retention of a form of error norm monotonicity is possible using a topologically equivalent norm , carefully chosen to match the structure of the algorithm. The robustness criteria derived extend the substantial NOILC robustness studies in [7] to the new algorithm and also provide robustness results for state space tracking and end-point problems. Frequency domain conditions for robustness can be derived for some cases of interest with the robustness condition for the "inverse algorithm" underlying all results. The robustness criteria reveal the trade off between "gain" and error correction showing how the combined effect can have a neutral effect on algorithm robustness.

The ideas generalize to include a finite number m_G of gradient computations between each NOILC iteration. This mechanism makes arbitrary convergence characteristics possible but at the expense of increased computational load. Furthermore, beyond ILC there are other methods with an iterative nature, e.g., reinforcement learning [37], Bayesian optimization [38]. Whether or not ideas from these methods can be applied in ILC, in particular, to accelerate the convergence, would be an interesting question. The above constitutes part of our future research and will be reported separately.

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. S. Ahn, Y. Chen, and K. L. Moore, "Iterative learning control: brief survey and categorization," *IEEE Transactions on Systems, Man and Cybernetics, Part C*, vol. 37, no. 6, pp. 1109–1121, 2007.

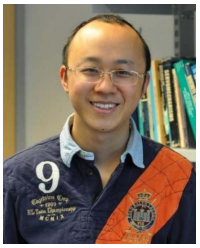
[3] K. L. Moore, *Iterative Learning Control for Deterministic Systems. Advances in Industrial Control.* Springer-Verlag, 1993.

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

[5] Z. Bien and J. Xu, *Iterative learning control: Analysis, integration, and application*. Kluwer Academic Publishers, 1998.
[6] J. Xu, S. Panda, and T. Lee, *Real-time iterative learning control: design and applications*. Springer-Verlag, 2009.
[7] D. Owens, *Iterative Learning Control: An Optimization Paradigm*. Springer, 2015.
[8] M. Norrlof, "An adaptive iterative learning control algorithm with experiments on an industrial robot," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 2, pp. 245–251, 2002.
[9] J. H. Lee and K. S. Lee, "Iterative learning control applied to batch processes: An overview," *Control Engineering Practice*, vol. 15, pp. 1306–1318, 2007.
[10] T. Liu and Y. Q. Wang, "A synthetic approach for robust constrained iterative learning control of piecewise affine batch processes," *Automatica*, vol. 48, no. 11, pp. 2762–2775, 2012.
[11] C. T. Freeman, *Control System Design for Electrical Stimulation in Upper Limb Rehabilitation*. Springer International Publishing, 2016.
[12] S. S. Saab, "Stochastic P-type/D-type iterative learning control algorithms," *International Journal of Control*, vol. 76, no. 2, pp. 139–148, 2003.
[13] C. Chien and A. Tayebi, "Further results on adaptive iterative learning control of robot manipulators," *Automatica*, vol. 44, no. 3, pp. 830—-837, 2008.
[14] R. W. Longman, Y. Peng, T. Kwon, H. Lus, R. Betti, and J. Juang, "Adaptive inverse iterative learning control," *Journal of the Chinese Society of Mechanical Engineers*, vol. 32, no. 6, pp. 493–506, 2011.
[15] J. Ghosh and B. Paden, "A pseudoinverse-based iterative learning control," *IEEE Transactions on Automatic Control*, vol. 47, no. 5, pp. 831–837, 2002.
[16] K. Kinoshita, T. Sogo, and N. Adachi, "A relationship between iterative learning control using the gradient method and stable inversion," *Transactions of the Society of Instrument and Control Engineers*, vol. 36, no. 12, pp. 1117–1124, 2000.
[17] D. H. Owens, J. Hatonen, and 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.
[18] N. Amann, D. H. Owens, and E. Rogers, "Iterative learning control using optimal feedback and feedforward actions," *International Journal of Control*, vol. 65, no. 2, pp. 277–293, 1996.
[19] S. Gunnarsson and M. Norrlof, "On the design of ILC algorithms using optimization," *Automatica*, vol. 37, no. 12, pp. 2011–2016, 2001.
[20] D. De Roover and O. H. Bosgra, "Synthesis of robust multivariable iterative learning controllers with application to a wafer stage motion system," *International Journal of Control*, vol. 73, no. 10, pp. 968–979, 2000.
[21] K. Moore, H. Ahn, and Y. Chen, "Iteration domain H_∞-optimal iterative learning controller design," *International Journal of Robust and Nonlinear Control*, vol. 18, no. 10, pp. 1001–1017, 2008.
[22] B. Cichy, K. Galkowski, and E. Rogers, "2D systems based robust iterative learning control using noncausal finite-time interval data," *Systems & Control Letters*, vol. 64, pp. 36–42, 2014.
[23] 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 Transactions on Mechatronics*, vol. 11, no. 1, pp. 55–65, 2006.
[24] C. T. Freeman and Y. Tan, "Iterative learning control with mixed constraints for point-to-point tracking," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 3, pp. 604–616, 2013.
[25] P. Janssens, G. Pipeleers, and 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.
[26] T. D. Son, H. S. Ahn, and K. L. Moore, "Iterative learning control in optimal tracking problems with specified data points," *Automatica*, vol. 49, no. 5, pp. 1465–1472, 2013.
[27] T. D. Son, G. Pipeleers, and J. Swevers, "Robust monotonic convergent iterative learning control," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, pp. 1063–1068, 2016.
[28] T. Oomen, J. van de Wijdeven, and O. H. Bosgra, "System identification and low-order optimal control of intersample behavior in ILC," *IEEE Transactions on Automatic Control*, vol. 56, no. 11, pp. 2734–2739, 2011.
[29] M. Togai and O. Yamano, "Analysis and design of an optimal learning control scheme for industrial robots: A discrete system approach," in *1985 24th IEEE Conference on Decision and Control*. IEEE, 1985, pp. 1399–1404.
[30] K. Furuta and M. Yamakita, "The design of a learning control system for multivariable systems," in *Proc. of IEEE Int. Symp. on Intelligent Control*, 1987.
[31] K. Kinoshita, T. Sogo, and N. Adachi, "Adjoint-type iterative learning control for a single-link flexible arm," *IFAC Proceedings Volumes*, vol. 35, no. 1, pp. 223–228, 2002.
[32] D. H. Owens and B. Chu, "Error corrected references and acceleration of norm optimal iterative learning control," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 5594–5599.
[33] B. Chu and 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.
[34] J.-X. Xu, Y. Chen, T. Lee, and S. Yamamoto, "Terminal iterative learning control with an application to RTPCVD thickness control," *Automatica*, vol. 35, no. 9, pp. 1535–1542, 1999.
[35] Y.Wang and Z. Hou, "Terminal iterative learning control based station stop control of a train," *International Journal of Control*, vol. 84, no. 7, pp. 1263–1277, 2011.
[36] C. T. Freeman, "Upper limb electrical stimulation using input-output linearization and iterative learning control," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 4, pp. 1546–1554, 2014.
[37] B. Recht, "A tour of reinforcement learning: The view from continuous control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, pp. 253–279, 2019.
[38] R. Garnett, *Bayesian optimization*. Cambridge University Press, 2023.

David H. Owens received the B.Sc. degree in theoretical physics and the Ph.D. degree in control engineering from Imperial College London, London, U.K, in 1969 and 1973, respectively. He is an author of several texts on multivariable, repetitive and iterative control and was elected a Fellow of the UK Royal Academy of Engineering in 2008.

He spent four years with the United Kingdom Atomic Energy Aithority (1969-73) and, subsequently, has held senior academic positions at the Universities of Strathclyde, Exeter and Sheffield, UK. Previously the Head of Department and Dean of the Faculty of Engineering, he is now a Professor Emeritus with the Department of Automatic Control and Systems Engineering, The University of Sheffield, U.K. He has worked, part-time, with the Italian Institute of Technology, Genoa, has been a Visiting Professor at the University of Southampton and was the Head of the Robotics Group in the School of Electrical Engineering, Zhengzhou University, China (2016-2021). His current research interests include the theoretical basis of, and algorithm development for, robust optimization-based iterative learning control and its applications .



Bing Chu received the B.Eng. degree in automation and the M.Sc. degree in control science and technology from Tsinghua University, Beijing, China, in 2004 and 2007, respectively, and the Ph.D. degree in automatic control and systems engineering from The University of Sheffield, Sheffield, U.K., in 2009.

He was a Post-Doctoral Researcher with the University of Oxford, Oxford, U.K., from 2010 to 2012. He is currently a Professor in electronics and computer science with the University of Southampton, Southampton, U.K. His current research interests include iterative learning and repetitive control, analysis and control of large-scale networked systems, applied optimization theory, and their applications to robotics, power electronics, and next-generation healthcare. Dr. Chu was a recipient of a number of awards, including the Best Paper Prize of the 2012 United Kingdom Automatic Control Council International Conference on Control and the Certificate of Merit for the 2010 IET Control and Automation Doctoral Dissertation Prize.