



Robust iterative learning control for unstable MIMO systems

Lucy Hodgins & Chris T. Freeman

To cite this article: Lucy Hodgins & Chris T. Freeman (03 Jun 2025): Robust iterative learning control for unstable MIMO systems, International Journal of Control, DOI: [10.1080/00207179.2025.2513674](https://doi.org/10.1080/00207179.2025.2513674)

To link to this article: <https://doi.org/10.1080/00207179.2025.2513674>



© 2025 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.



Published online: 03 Jun 2025.



Submit your article to this journal [↗](#)



Article views: 209



View related articles [↗](#)



View Crossmark data [↗](#)

Robust iterative learning control for unstable MIMO systems

Lucy Hodgins and Chris T. Freeman

Electronics and Computer Science, University of Southampton, Southampton, UK

ABSTRACT

Iterative learning control (ILC) is a well-established technique to successively improve tracking accuracy for systems that repeatedly perform the same task. Most current literature imposes constraints on the nature of the system, such as requiring it to be full-rank, or inherently stable. This paper presents a generalised ILC framework that can handle non-linear, unstable, MIMO systems with rank deficiency. This involves the minimisation of a cost function that balances tracking performance and input effort, extending previous approaches to include a 'robustness filter' within the optimisation. Gap metric analysis is then applied to examine the robustness of the resulting system, with performance bounds derived for both serial and parallel ILC architectures. A design procedure is presented that allows the designer to transparently trade-off robustness and convergence properties. The design framework is illustrated via application to the inverted pendulum problem, a classic example of a highly nonlinear, unstable, and under-actuated system.

ARTICLE HISTORY

Received 2 September 2024
Accepted 26 May 2025

KEYWORDS

Iterative learning control;
robust control; inverted
pendulum; multiple
input-multiple-output;
optimisation-based design

1. Introduction

Iterative learning control (ILC) is a control method designed to enable high tracking accuracy when carrying out a task repeatedly over a fixed time interval. This is achieved by adjusting the input on each iteration using error information gained from one or more previous trials. First formulated in Arimoto et al. (1984), there now exist a wide breadth of algorithms specific to particular systems and applications (Chi et al., 2013; Hazarika & Swarup, 2020; Xu et al., 2020; Yu et al., 2018). A standard assumption of ILC is that the tracking task is achievable, and dynamics are often constrained to be linear SISO with relative degree one. Recent research has broadened this to remove the necessity for the system to have full rank, thereby enabling general MIMO classes with rank deficiency (Chen, 2022). ILC has been used extensively with non-linear systems, and a popular approach involves linearising the system about a suitable operating point and then applying linear ILC design. The use of a model improves convergence speed when compared to model-free designs (Owens, 2016), however the mismatch between the model dynamics and those of the true plant can result in instability. The standard approach used in ILC to tackle modelling error is to introduce a so-called 'robustness filter' or 'Q-filter' which takes the form of a zero-phase filter applied to the control input at every trial (Bristow et al., 2006). This is used by the majority of practitioners, but no attempt to optimise its

design (i.e. formulate the filter design within an optimisation problem involving robust performance) has yet been proposed.

The robustness of ILC to modelling error has been studied using a variety of techniques (Ahn et al., 2005; Donkers et al., 2008; Freeman et al., 2009; Hao et al., 2019; Owens et al., 2014; Qian et al., 2021; Xu & Xu, 2013). To avoid the assumption of highly structured uncertainty, Bradley (2010) employed the gap metric, building on the framework of Georgiou and Smith (1997). These results precisely bounded the distance between plant model and the true plant in order to maintain robust performance. Results were generalised further in Freeman (2017) to include ILC in combination with a feedback controller. However, these analyses make the assumption that the plant is inherently stable, preventing their applicability to a wider range of practical applications. These include the control of quadcopters, as well as the inverted pendulum problem, which has parallels between the control of both bipedal walking and a rocket at takeoff.

The inverted pendulum or 'cart-pole' problem is a common benchmarking problem and typically involves controlling a cart with a pendulum attached, by adjusting the input to the wheels of the cart such that the pendulum remains in the upright position and the cart tracks a fixed trajectory. This results in a system that is highly nonlinear, under-actuated, and inherently unstable. There have been various attempts to apply model-based ILC to the

CONTACT Lucy Hodgins  lgh1g19@soton.ac.uk  Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK

© 2025 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

inverted pendulum problem (Binz & Aronovskiy, 2021; Meindl et al., 2024; Precup et al., 2009; Zhan, 2010), however these rarely consider control of both the pendulum angle and position. Those that do make use of linearised models of the system for design without considering the implications that this may have on system robustness.

This paper makes several contributions to robust ILC design theory and practice: first it proposes a new design objective that is suitable for a wide class of MIMO, rank deficient systems. It extends the work of Chen and Freeman (2019) by posing ILC as an optimisation problem, but relaxes the requirement that tracking error be minimised in order to yield far more robust solutions. This generalised ILC framework also lifts restrictions on the form of the learning operator. Furthermore, it establishes that the ubiquitous ‘Q-filter’ can be transparently formulated as the solution to an optimisation problem. Secondly, robust performance results are derived by extending the work of Bradley (2010) and Freeman (2017) to allow for application to unstable plants, and both forms of ILC architecture (serial and parallel). A principled design procedure is then proposed which enables the designer to transparently trade robustness with convergence accuracy and speed for the first time. Finally, the design approach is demonstrated via application to an inverted pendulum.

This paper is structured as follows. Section 2 introduces a new framework for ILC convergence analysis which embeds a robustness filter. Section 3 then analyses the robust performance of the system, before a design procedure is introduced in Section 4. This is then applied to the inverted pendulum system in Section 5, before Section 6 concludes the paper.

2. Problem formulation

Let G^* represent the true, possibly non-linear system to be controlled, expressed in the form of the discrete-time operator

$$G^* : \ell_2^q[0, T] \rightarrow \ell_2^p[0, T] : u_k \mapsto y_k. \quad (1)$$

where q and p are the number of inputs and outputs respectively. In ILC it is assumed that the system repeatedly performs the same task, over a fixed time duration $T < \infty$. Each iteration is termed a ‘trial’, and its signals are denoted by the subscript $k \in \mathbb{N}_+$. It is also assumed that the system is initiated from identical initial conditions. In contrast to previous studies (Freeman, 2017), no assumption is placed on the stability of this system, neither is $G^*(0) = 0$ assumed. It is common practice in ILC to apply a feedback controller around the system to ensure reasonable tracking performance on the first trial, and mitigate the effects of noise and disturbance.

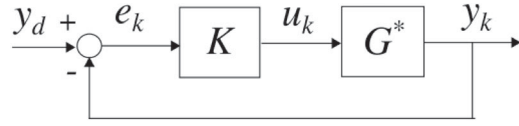


Figure 1. Feedback control system $[G^*, K]$.

The resulting arrangement is shown in Figure 1, with feedback controller operator

$$K : e_k \mapsto u_k, \quad (2)$$

where $e_k = y_d - y_k$ is the tracking error. It will be assumed that K satisfies $G^*K(0) = 0$ and the closed loop $[G^*, K]$ is well-posed.

The standard control problem is for the system output y_k to track a reference signal $y_d \in \ell_2^p[0, T]$ as $k \in \mathbb{N}_+$ increases. In ILC, this is achieved by successively updating a feedforward signal v_k that is added to the closed-loop system. This can be realised in one of two standard configurations (Bristow et al., 2006), as shown in Figure 2.

The standard ILC objective is to design v_k such that the update and output converge as

$$\lim_{k \rightarrow \infty} v_k = v_\infty, \quad \lim_{k \rightarrow \infty} y_k = y_d$$

which implies $\lim_{k \rightarrow \infty} e_k = 0$. In cases where this objective is feasible, it can be achieved using the ILC update structure

$$v_{k+1} = Q(v_k + Le_k), \quad k = 0, 1, \dots, v_0 = 0 \quad (3)$$

in which learning operator L is given by

$$L : \ell_2^p[0, T] \rightarrow \ell_2^q[0, T] : e_k \mapsto v_k, \quad (4)$$

$$L : \ell_2^p[0, T] \rightarrow \ell_2^q[0, T] : e_k \mapsto v_k, \quad (5)$$

for the parallel and serial cases respectively, both with $L(0) = 0$. The differing definitions of L are due to the dimensions of v_k being different in each case. Note however that the generalised results outlined in this paper hold for both definitions of operator L , and thus we will not explicitly differentiate between the two forms. The operator Q is a so-called ‘robustness filter’ which is used to trade tracking performance for robustness. As previously discussed, ILC algorithms are often designed based on a linear approximation

$$G : \ell_2^q[0, T] \rightarrow \ell_2^p[0, T] : u_k \mapsto y_k \quad (6)$$

to the true plant dynamics $G^* : \ell_2^q[0, T] \rightarrow \ell_2^p[0, T] : u_k \mapsto y_k$, as this simplifies convergence properties and allows for the derivation of precise optimality conditions. These linearisations are designed about a particular operating point, for example making use of the Jacobian of the system, or using small angle approximations.

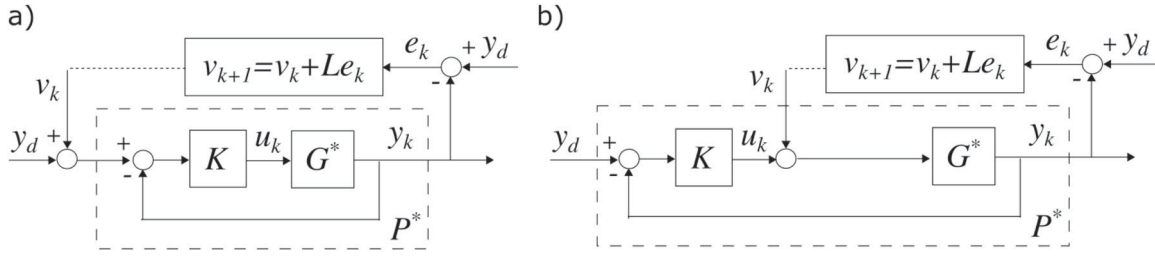


Figure 2. ILC with conventional (a) serial and (b) parallel feedback architectures. In each case the closed loop $[G^*, K]$ will be denoted P^* .

The following section now shows how L can be derived as the solution of a general optimisation problem involving the linearised system G and feedback controller K . It extends the framework of Chen and Freeman (2019) by adding the feedback controller and generalising the ILC update law. Note that this section is concerned with the design of controllers based on the linearised system G . The robustness of this controller when applied to the true non-linear system G^* is examined in Section 3.

2.1 Design framework

As discussed, the standard ILC objective (3) implies that the reference can be tracked perfectly. Furthermore, existing updates generally assume that the corresponding converged ILC input v_∞ is unique. As a result, standard ILC algorithms cannot be used with MIMO systems where $q \neq p$, or those with restrictive constraints, e.g. non-fully functional, rank deficient, input-output delayed or non-communicable dynamics. The most general existing formulation, introduced in Chen and Freeman (2019), addressed this by defining a generalised ILC objective involving both the tracking error and input signal. The next definition extends this framework to add the feedback controller K , enabling application to open-loop unstable systems.

Definition 2.1: Consider the configurations shown in either Figure 2(a,b), where G^* is replaced by nominal system G , and K is designed to stabilise the closed-loop system $[G, K]$. The *Generalised ILC Objective* is to design operators L and Q in the update form (3) such that ILC converges to minimise the tracking error norm, i.e

$$\lim_{k \rightarrow \infty} y_k = y^*,$$

$$y^* := \arg \min \{J(y_k) \mid J(y_k) = \|y_d - y_k\|^2\} \quad (7)$$

In addition, the input signal must converge to the optimal solution

$$\lim_{k \rightarrow \infty} v_k = v^*,$$

$$v^* = \arg \min \{J(v_k) \mid J(v_k) = \|v_k\|_W^2, y_k = y^*\} \quad (8)$$

where W is a symmetric positive definite weighting operator.

The Generalised ILC Objective allows for a trade-off between tracking performance and input energy, with the balance depending on the weighting operator W that is selected by the designer. It makes no assumptions regarding the system rank, allowing it to be applied to over- or under-actuated systems. The following theorem now defines the operators L and Q that solve the Generalised ILC Objective.

Theorem 2.2: Denote the closed-loop system $[G, K]$ by operator P , such that $P := (I + GK)^{-1}GK$, $P := (I + GK)^{-1}G$ for the serial and parallel configurations respectively, as shown in Figure 3. Then the Generalised ILC Objective is satisfied if $Q = I$, and L is selected such that PL is symmetric positive semi-definite with $\text{rank}(PL) = \text{rank}(P)$ and $\rho(PL) < 2$. In particular, properties (7), (8) hold with the weight W satisfying

$$L = W^{-1}P^\top. \quad (9)$$

Furthermore, the ILC sequence will achieve monotonic reduction in tracking error norm, i.e.

$$\|e_{k+1}\|^2 \leq \|e_k\|^2 \quad \forall k \in \mathbb{N}_+. \quad (10)$$

Proof: Minimising error norm (7) for arbitrary y_d requires that $Q = I$. The requirement for L follows by adapting Theorem 1 of Chen (2022) and replacing G by each of the closed loop system P representations. ■

2.2 Robust design framework

Theorem 2.2 defines an optimal ILC update for a general class of system. Unfortunately the requirement that an arbitrary reference y_d be tracked as closely as possible meant that it was necessary to select $Q = I$, and it is well known that the resulting ILC update form

$$v_{k+1} = v_k + Le_k \quad (11)$$

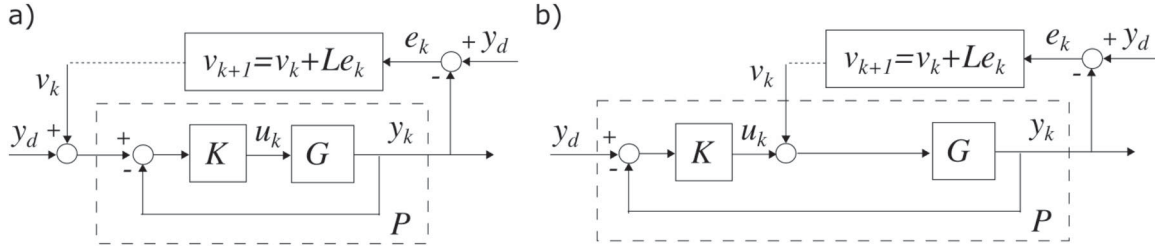


Figure 3. ILC with (a) serial and (b) parallel feedback architectures. In each case the closed loop $[G, K]$ is denoted P .

has very limited robustness. For example, it cannot tolerate a model phase uncertainty of over 90 degrees in magnitude without becoming unstable (Freeman et al., 2009). As previously stated, the common approach in ILC to address this is to choose Q as a so-called ‘robustness filter’ which typically takes the form of zero-phase low pass filter (Freeman et al., 2009). While robustness filter Q is routinely used in ILC, its design has always been ad hoc, and based on the selection of a suitable cut-off frequency. Its inclusion destroys the optimal tracking accuracy condition of (7). More generally, the filter has never been incorporated in an optimisation problem which would allow its selection to rigourously balance the competing demands of tracking, convergence speed and robustness to model uncertainty.

The next result addresses this limitation by showing that the ILC update form (3) is the iterative solution to a minimisation problem which balances both (weighted) tracking accuracy and (weighted) control effort. In particular, it defines a new cost function which takes the tracking requirement of (7) and adds the (weighted) control effort term in (8). Since these are now in a single cost function, the minimisation relaxes the requirement to primarily focus on accuracy. It will be shown in Section 3 that this has a substantial effect on overall robustness to modelling error when applied to the true nonlinear system.

Definition 2.3: Consider the two possible ILC configurations shown in Figure 3(a,b), where K is designed to stabilise the closed-loop systems $P := (I + GK)^{-1}GK$, $P := (I + GK)^{-1}G$ respectively. The *Robust Generalised ILC Objective* is to design operators Q and L in the update form (3) such that ILC converges to minimise the combined tracking error norm and control effort norm, i.e

$$\begin{aligned} \lim_{k \rightarrow \infty} y_k &= y^*, \quad \lim_{k \rightarrow \infty} v_k = v^*, \\ (y^*, v^*) &= \arg \min \{J(y_k, v_k) \mid J(y_k, v_k) \\ &= \|y_d - y_k\|_X^2 + \|v_k\|_Z^2\} \end{aligned} \quad (12)$$

where X and Z are symmetric positive definite and positive semi-definite weighting operators respectively.

The following theorem now outlines how operators Q and L can be designed to satisfy this new robust objective.

Theorem 2.4: *The Robust Generalised ILC Objective is satisfied if ILC learning operator L is selected such that PL is symmetric positive semi-definite with $\text{rank}(PL) = \text{rank}(P)$, and Q is a multi-channel zero-phase filter satisfying*

$$\|Q(I - LP)\| := \zeta < 1. \quad (13)$$

In particular, the input and output signals will converge to the optimal solution given by (12) with the weighting operators Z and X given by

$$\begin{aligned} Z &= I - Q, \\ P^T X &= QL. \end{aligned} \quad (14)$$

Furthermore, the ILC sequence will achieve monotonic reduction in tracking error norm, i.e.

$$\|\tilde{e}_{k+1}\|^2 \leq \zeta \|\tilde{e}_k\|^2 \quad \forall k \in \mathbb{N}_+. \quad (15)$$

where $\tilde{e}_k = y_{d,k} - y_k$ is the tracking error with respect to the achievable component of the reference y_d , defined as

$$y_{d,k} = y_d - \sum_{i=0}^{k-1} (Q(I - PL))^i (I - Q)(I + P)y_d. \quad (16)$$

Proof: Consider the serial case: substitute $y_k = P(v_k + y_d)$ and expand (12) as

$$\begin{aligned} J(y_k, u_k) &= \|y_d - y_k\|_X^2 + \|v_k\|_Z^2 \\ &= \|y_d - P(v_k + y_d)\|_X^2 + \|v_k\|_Z^2 \end{aligned}$$

Differentiating wrt v_k and setting equal to zero yields

$$\begin{aligned} 0 &= -P^T X((I - P)y_d - P v_k) + Z v_k \\ \Rightarrow v^* &= (Z + P^T X P)^{-1} P^T X (I - P) y_d \end{aligned} \quad (17)$$

Substituting and expanding robust update law (3) gives

$$v_{k+1} = Q(v_k + L(y_d - P(v_k + y_d)))$$

$$= (Q(I - LP))^k v_0 + \sum_{i=0}^{k-1} (Q(I - LP))^i QL(I - P)y_d \quad (18)$$

so that, since (13) holds by assumption and $v_0 = 0$, we get

$$v^* := \lim_{k \rightarrow \infty} v_k = (I - Q(I - LP))^{-1} QL(I - P)y_d. \quad (19)$$

Comparison with (17) then yields the required (14) relations, where existence of L is guaranteed by the assumption that PL is symmetric positive semi-definite. The corresponding error evolution is

$$\begin{aligned} e_{k+1} &= y_d - P(y_d + Q(v_k + Le_k)) \\ &= y_d - Py_d - PQv_k - PQLe_k \end{aligned}$$

and the structure of Q produces

$$\begin{aligned} e_{k+1} &= y_d - Py_d - PQv_k - PQLe_k \\ &= y_d - (I - Q)Py_d - Qy_k - PQLe_k \\ &= Qy_d - (I - Q)y_d - (I - Q)Py_d - Qy_k - PQLe_k \\ &= Qe_k - (I - Q)y_d - (I - Q)Py_d - PQLe_k \\ &= Q(I - PL)e_k - (I - Q)(I + P)y_d. \end{aligned} \quad (20)$$

The achievable reference (16) satisfies the dynamic relation

$$\begin{aligned} (y_d - y_{d,k+1}) &= Q(I - PL)(y_d - y_{d,k}) \\ &\quad - (I - Q)(I + P)y_d, \end{aligned} \quad (21)$$

and subtracting from (20) gives

$$\tilde{e}_{k+1} = Q(I - PL)\tilde{e}_k. \quad (22)$$

Applying the triangle inequality and bound (13) gives (15) as required. The proof for the parallel case is identical but with $(v_k + y_k)$ replaced with $(v_k + Ky_d)$. ■

Remark 2.1: The output of the closed-loop system converges to

$$y^* = P((I - Q(I - LP))^{-1} QL(I - P) + I)y_d, \quad (23)$$

$$y^* = P((I - Q(I - LP))^{-1} QL(I - PK) + K)y_d, \quad (24)$$

for the serial and parallel case respectively, with corresponding error given by

$$\begin{aligned} e_\infty &= (I - P((I - Q(I - LP))^{-1} QL(I - P) + I))y_d, \\ e_\infty &= (I - P((I - Q(I - LP))^{-1} QL(I - PK) + K))y_d. \end{aligned} \quad (25)$$

These solve the minimisation problem (12).

Theorem 2.4 enables the designer to trade ILC accuracy with control effort by selecting L and Q to generate the required weights in (12). i.e. to realise the desired weights X, Z , the designer must select

$$\begin{aligned} Q &= I - Z, \\ L &= Q^{-1}P^\top X. \end{aligned} \quad (26)$$

Remark 2.2: Setting $Z = 0$, yields the minimum error solution, with the new ILC objective (12) reducing to the previous cost function form of (7). This is achieved using the selection

$$\begin{aligned} Q &= I, \\ L &= P^\top X. \end{aligned} \quad (27)$$

If X is chosen as a scalar, $\beta \in \mathbb{R}_+$, this corresponds to the well-known gradient ILC algorithm (see. e.g. Chen & Freeman, 2019; Freeman et al., 2009). The plant output (23), (24) then equals $y^* = PP^\dagger y_d$, where $P^\dagger$ is the pseudo-inverse of P (Chen & Freeman, 2019), which is an orthogonal projection on the range of P . Unfortunately, the gradient ILC algorithm is well-known to be slow to converge, which can be confirmed by inspecting the error convergence expression (15) and noting that the 2-norm of $Q(I - LP) = I - \beta P^\top P$ is approximately unity for many systems. To understand why this is the case, the convergence rate at a specific frequency, ω , can be ascertained by computing the magnitude of this operator in the frequency domain for each input-output pair, i.e. $|1 - \beta|P(\omega)|^2|$. Since for most electromechanical systems $|P(\omega)| \rightarrow 0$ as $\omega \rightarrow \infty$, it follows that $|1 - \beta|P(\omega)|^2| \rightarrow 1$ so that convergence tends to zero at high frequencies.

To increase the convergence rate at higher frequencies, it is therefore sensible to select the weight X in cost function (12) to amplify high frequencies in inverse proportion to their attenuation by the plant. This motivates choosing $X = (PP^\top)^{-1}$, which has a frequency magnification of $\frac{1}{|P(\omega)|^2}$. This generates the ILC update term $QL = \beta(I + qPP^\top)^{-1}$, which corresponds to inverse ILC (Owens et al., 2014). Here $\beta > 0$ has been added to provide additional control over the convergence at all frequencies. Further control over the convergence rate at higher frequencies is obtained by introducing a parameter q and modifying the weight to $X = (I + qPP^\top)^{-1}$. This has a frequency magnification of $\frac{1}{1 + q|P(\omega)|^2}$ and allows the designer to reduce the amplification of the high-frequency components, thereby providing more flexibility over the convergence rate at these frequencies. The ILC update term $QL = \beta P^\top (I + qPP^\top)^{-1}$ corresponds to the norm-optimal ILC algorithm (Amann

Table 1. Design choices for weights in Theorem 2.4, with resulting ILC operators.

Desired properties		Weights in cost function		ILC update terms	
Tracking magnifier	Input range	X	Z	Q	QL
1	all w	β	0	I	βP^T
$\frac{1}{ P(w) ^2}$	all w	$\beta (PP^T)^{-1}$	0	I	$\beta P^T (PP^T)^{-1}$
$\frac{q}{1 + q P(w) ^2}$	all w	$\beta (I + qPP^T)^{-1}$	0	I	$\beta P^T (I + qPP^T)^{-1}$
1	$w \in [0, w_c]$	β	HPF ¹	LPF ²	βP^T
$\frac{1}{ P(w) ^2}$	$w \in [0, w_c]$	$\beta (PP^T)^{-1}$	HPF	LPF	$\beta P^T (PP^T)^{-1}$
$\frac{q}{1 + q P(w) ^2}$	$w \in [0, w_c]$	$\beta (I + qPP^T)^{-1}$	HPF	LPF	$\beta P^T (I + qPP^T)^{-1}$

Note: ¹HPF denotes zero-phase high-pass filter, cutoff w_c .

²LPF denotes zero-phase low-pass filter, cutoff w_c .

et al., 1996; Bristow, 2008). Again, the parameter β allows for additional tuning of convergence speed over all frequency components.

As well as selecting X based on desired tracking and convergence properties, the designer may want to choose weight Z to reduce high frequencies in the input signal, since these are associated with actuator wear and noise corruption. This is easily done by setting Z to realise a zero phase high-pass filter, with cut-off frequency selected as the highest frequency component of the reference. This corresponds to Q taking the form of a low-pass filter, as is traditionally implemented in standard ILC to obtain improved robustness. Based on these requirements, Table 1 summarises common design choices for the weights X and Z , together with the resulting operators Q , QL that are needed in the ILC update.

2.3 Alternative robust design framework

Theorem 2.4 demonstrates that the ILC update law (3) converges to an optimal solution of the expanded cost function (12), which comprises the weighted sum of the norms of the error and the input v_k to the closed-loop plant. However, in some cases the designer may wish to minimise the input to the plant, u_k , rather than the ILC control signal. The following theorem therefore extends the above result to allow the designer to exchange v_k for u_k within cost function (12), hence balancing control effort against tracking error.

Theorem 2.5: Suppose the plant G is full rank with $p \geq q$, and the reference satisfies $Qy_d = y_d$. Then Theorem 2 holds with the cost function (12) replaced by

$$\begin{aligned} \lim_{k \rightarrow \infty} y_k &= y^*, \quad \lim_{k \rightarrow \infty} v_k = v^*, \\ (y^*, v^*) &= \arg \min \{J(y_k, v_k) \mid J(y_k, v_k) \\ &= \|y_d - y_k\|_X^2 + \|u_k\|_W^2\} \end{aligned} \quad (28)$$

provided the weights Z and W satisfy

$$Z = P^T (G_L^{-1})^T W G_L^{-1} P \quad (29)$$

where G_L^{-1} is any left inverse of plant operator G .

Proof: The assumptions on G mean there exists a symmetric operator $A = (G_L^{-1})^T W G_L^{-1}$. Using this operator, we can write $W = G^T (G_L^{-1})^T W G_L^{-1} G = G^T A G$. Substituting this for W in cost function (28) gives

$$\arg \min \{J(y_k, v_k) \mid J(y_k, v_k) = \|y_d - y_k\|_X^2 + \|y_k\|_A^2\}$$

since $y_k = Gu_k$. In the serial case this equates to

$$\begin{aligned} \arg \min \{J(y_k, v_k) \mid J(y_k, v_k) \\ = \|y_d - y_k\|_X^2 + \|P(v_k + y_d)\|_A^2\} \end{aligned}$$

Setting $Z = P^T A P$ means this is equivalent to

$$\arg \min \{J(y_k, v_k) \mid J(y_k, v_k) = \|y_d - y_k\|_X^2 + \|v_k + y_d\|_Z^2\} \quad (30)$$

whose solution is

$$\begin{aligned} v^* &= (I - Q(I - LP))^{-1} QL(I - P)y_d \\ &\quad - (I - Q(I - LP))^{-1} (I - Q)y_d \end{aligned} \quad (31)$$

If $Qy_d = y_d$, then $(I - Q)y_d = 0$ and this equals (17) which is the solution when applying Theorem 2.4. The parallel case follows using similar steps. ■

Theorem 2.5 allows the minimisation term $\|v_k\|_Z^2$ in cost function (12) to be exchanged for $\|u_k\|_W^2$ simply by setting Z equal to (29). Note that the condition of $Qy_d = y_d$ can be ensured through the selection of Z as a high-pass filter with a cut-off frequency above those frequencies present in the reference, as was proposed in Table 1.

3. Robustness analysis

The previous section derived performance criteria for ILC update law (3) when applied to the closed-loop system configurations shown in Figure 3. These involve the approximate model G and resulting closed loop system $[G, K]$, which was denoted P . However in practice ILC will be applied to the true system shown in Figure 2, in which G^* is the non-linear plant and the closed loop system $[G^*, K]$ is denoted P^* . This section therefore derives conditions on the allowable mismatch that can exist between P and P^* whilst maintaining stability. The results derived in this section extend those considered in Bradley (2010) and Freeman (2017) since they embed the feedback controller, multiple configurations, and remove the requirement that G^* is stable.

The gap metric, $\vec{\delta}$, is used and is a common measure of distance between two systems that appears widely in robust control (Qiu & Zhao, 2021). The seminal work in Georgiou and Smith (1997) considered the standard feedback forms shown in Figure 4, in which H^* is the true plant and H is a model of H^* used to design a feedback controller C .

The framework employs external disturbances represented by signals y_0 and u_0 , and external biases represented by $\bar{y}_0$ and $\bar{u}_0$. Gap-based robustness analysis is concerned with determining whether a controller designed for model H will stabilise the true plant H^* . To do this a projection operator is defined as the mapping from external to internal signals of the model system $[H, C]$, i.e.

$$\Pi_{H/C} : \begin{pmatrix} u_0 + \bar{u}_0 \\ y_0 + \bar{y}_0 \end{pmatrix} \rightarrow \begin{pmatrix} u_1 \\ y_1 \end{pmatrix} \quad (32)$$

The biased norm¹ of this projection operator, $\|\Pi_{H/C}\|_{(\bar{u}_0, \bar{y}_0)^\top}^\top$, characterises the performance of the closed-loop $[H, C]$, and the system is said to be gain-stable if it is bounded. It was shown that the closed-loop system $[H^*, C]$ will be stable if

$$\vec{\delta}(H, H^*) < \|\Pi_{H/C}\|_{(\bar{u}_0, \bar{y}_0)^\top}^{-1}. \quad (33)$$

To apply this analysis, it is first necessary to add disturbance signals, $u_{0,k}$, $y_{0,k}$, to the ILC systems shown in

Figure 2, giving rise to the systems shown in Figure 5. However, these systems do not coincide with the forms used in existing robustness analysis (i.e. Figure 4) since ILC operates in a trial to trial feedforward manner, rather than as conventional feedback system.

This can be addressed however by formulating the system dynamics so that each trial of the system is treated as a single sample. It is also necessary to define H and C to coincide with the two configurations under consideration. Using these techniques it is then possible to apply the gap analysis to ILC. The main robustness result is given in the next theorem.

Theorem 3.1: Let K be designed to stabilise the nominal closed-loop feedback system $[G, K]$, which is denoted P . Let ILC operators L, Q be selected to satisfy the conditions of Theorem 2.4. Suppose the controller is then applied to the true plant G^* , in either of the configurations shown in Figure 5, generating a closed-loop system $[G^*, K]$ denoted P^* . Then these systems are BIBO stable if the biased gap satisfies

$$\vec{\delta}(P, P^*) < b_{P,C}^{-1} \quad (34)$$

where

$$b_{P,C} = \left\| \begin{pmatrix} I \\ P \end{pmatrix} \right\| \left(1 + \frac{\|QLP\| + \|QL\|}{1 - \zeta} \right) \quad (35)$$

and the true plant signals are bounded with respect to their ideal values as

$$\|(v, y)^\top\|_{(v^*, y^*)^\top} \leq b_{P,C} \frac{1 + \vec{\delta}(P, P^*)}{1 - b_{P,C} \vec{\delta}(P, P^*)} \|(u_0, y_0)^\top\|. \quad (36)$$

Proof: Adding disturbance signals, $u_{0,k}$, $y_{0,k}$, to the ILC updates shown in Figure 2, results in the system shown in Figure 5. The system must then be arranged to resemble a standard feedback structure, that operates continually with no resetting. This is done by packaging each trial $t \in [0, T]$ as a single sample k of a higher dimensional ‘lifted’ system. For the serial case, the signals appearing in Figure 5(a) are therefore written as

$$v_k = v(k), \quad w_k = w(k), \quad e_k = e(k),$$

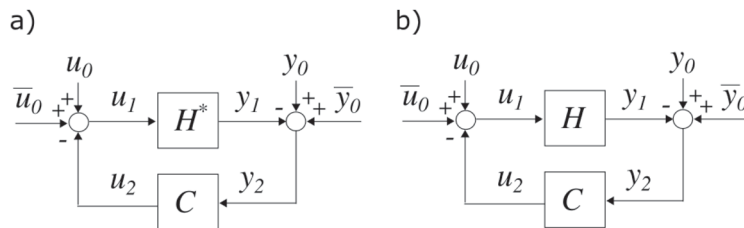


Figure 4. Closed loop feedback systems used within standard robust control theory: (a) true system $[H^*, C]$ and (b) model system $[H, C]$.

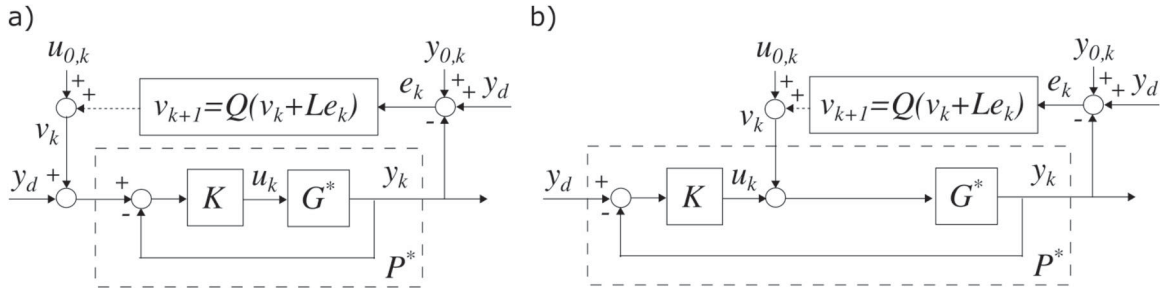


Figure 5. (a) Serial and (b) parallel architectures using true plant G^* with disturbances $u_{0,k}, y_{0,k}$ of compatible dimension.

$$y_k = y(k) \in \ell_2^p[0, T], \quad u_k = u(k) \in \ell_2^q[0, T] \quad (37)$$

and the corresponding lifted signal spaces are then defined as $v, e, w, y \in \ell_2^p[0, T] \times \mathbb{N}$. Define the lifted counterpart of trial-to-trial operator P as

$$\begin{aligned} \bar{P} : \ell_2^p[0, T] \times \mathbb{N} &\rightarrow \ell_2^p[0, T] \times \mathbb{N} : w \mapsto y \\ : y(k) &= Pw(k) \end{aligned} \quad (38)$$

with $\bar{P}^*$ defined similarly. Then defining the lifted counterpart of the ILC update as

$$\begin{aligned} \bar{C} : \ell_2^p[0, T] \times \mathbb{N} &\rightarrow \ell_2^p[0, T] \times \mathbb{N} : e \mapsto v \\ : v(k+1) &= Q(v(k) + Le(k)), \quad v(0) = 0. \end{aligned} \quad (39)$$

and setting $u_0 = u_{0,k}$ means that Figure 5(a) is equivalently represented as Figure 4(a), where H is substituted by P . Once the system is in this standard form, the results from Bradley (2010) can be applied, which bound the projection operator $\|\Pi_{\bar{P}/\bar{C}}\|_{(y_d, y_d)}^{-1}$ by the RHS of (34). Combining this with the bound

$$\vec{\delta}(\bar{P}, \bar{P}^*) < \|\Pi_{\bar{P}/\bar{C}}\|_{(y_d, y_d)}^{-1} \quad (40)$$

and with the result from Bradley (2010) that

$$\vec{\delta}(\bar{P}, \bar{P}^*) \leq \vec{\delta}(P, P^*) \quad (41)$$

yields (34) as required.

From Georgiou and Smith (1997), the projection norm of the true closed-loop system can be bounded as

$$\begin{aligned} \|\Pi_{\bar{P}^*/\bar{C}}\|_{(y_d, y_d)} &\leq \|\Pi_{\bar{P}/\bar{C}}\|_{(y_d, y_d)} \\ &\times \frac{1 + \vec{\delta}(P, P^*)}{1 - \|\Pi_{\bar{P}/\bar{C}}\|_{(y_d, y_d)} \vec{\delta}(P, P^*)}. \end{aligned} \quad (42)$$

which is then used to bound the internal signals of the system $[\bar{P}^*, \bar{C}]$ as

$$\|(v, y)^\top\|_{(v^*, y^*)} \leq \|\Pi_{\bar{P}^*/\bar{C}}\|_{(y_d, y_d)}^\top \|(u_0, y_0)^\top\|. \quad (43)$$

For the parallel case, the plant and controller of Figure 5(b) must be augmented as shown in Figure 6(b)

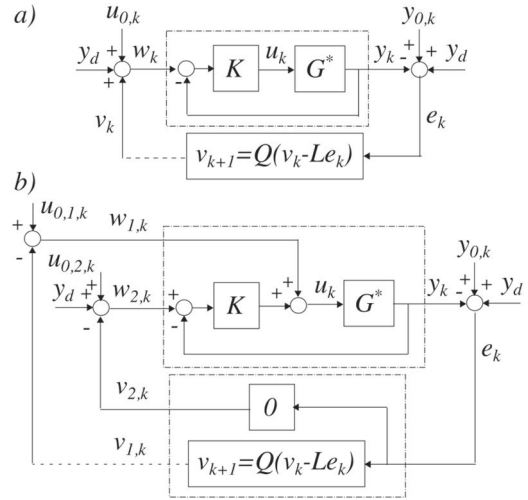


Figure 6. Augmented (a) serial and (b) parallel architectures.

in order to assume the structure of Figure 4(b). Stability of the augmented system guarantees stability of the original since the latter is a special case ($u_{0,2} = 0$). The augmented signals are $w_k = (w_{1,k}, w_{2,k})^\top$, $v_k = (v_{1,k}, v_{2,k})^\top$ and $u_{0,k} = (u_{0,1,k}, u_{0,2,k})^\top$, with the augmented lifted plant and ILC operators

$$\bar{P} : \ell_2^{q+p}[0, T] \times \mathbb{N} \rightarrow \ell_2^p[0, T] \times \mathbb{N} : w \mapsto y$$

$$: y(k) = (I + \bar{G}\bar{K})^{-1}\bar{G}(I, \bar{K})w(k),$$

$$\bar{C} : \ell_2^p[0, T] \times \mathbb{N} \rightarrow \ell_2^{q+p}[0, T] \times \mathbb{N} : e \mapsto v$$

$$: v(k+1) = \begin{pmatrix} 0 \\ Q((0, I)v(k) + Le(k)) \end{pmatrix}, \quad v(0) = 0.$$

respectively, and $\bar{P}^*$ defined similarly. Applying these signals and operators in relations (32)–(41) leads to the same overall bound (34) as in the serial case. ■

Condition (34) states that the ILC law stabilises a ‘ball’ of plants in the uncertainty space centred about the nominal system model P . The ball’s radius increases as $\|QLP\|$, $\|QL\|$ and ζ are reduced, where these quantities each have a direct relation to weighting operators X and Z

through (14). This means Theorem 3.1 provides a transparent method to design the ILC controller components to weigh performance against robustness. Computation of the biased gap $\vec{\delta}(P, P^*)$ can be simplified using the relation

$$\vec{\delta}(P, P^*) \leq \sup_{\|w\| \neq 0, k \in \mathbb{N}_+} \frac{\|(P^*|_{w_k} - P)w\|}{\|w\|}. \quad (44)$$

which is adapted from Freeman (2017). Here $P^*|_{w_k} w := P^*(w + w_k) - P^*(w_k)$ denotes the dynamics of P^* about the operating point used on ILC trial k (where $w_k = v_k + y_d$, $w_k = v_k + Ky_d$ for the serial and parallel cases respectively). It follows that the right-hand side of (44) can replace $\vec{\delta}(P, P^*)$ in (34), to provide a simple bound on the accuracy of the linearisation used to approximate the true plant dynamics.

Furthermore, the bound (36) explicitly demonstrates the influence of the disturbances (as measured by $\|(u_0, y_0)^\top\|$) and model uncertainties (as measured by the gap δ) on the tracking performance. Note in particular that the stability of the system, as determined by (34), is unaffected by the size of the disturbance, which purely affects tracking accuracy.

4. Design procedure

The results in the previous sections show the competing demands of maximising tracking accuracy (12), convergence speed (13) and robustness (34). The following steps are proposed in order to allow the designer to reach a transparent trade-off.

Step 1 – feedback controller K : Given a desired feedback architecture, design a feedback controller K to stabilise the true plant G^* about an operating point. Compute the corresponding linear model, G , about the desired operating point, as well as closed-loop dynamics P .

Step 2 – identify desired properties: Based on the intended application, identify the desired performance criteria, in terms of the range of frequencies that must be tracked and how fast convergence must be at each frequency. Also identify the required range of input frequencies that is permitted.

Step 3 – select weights X , Z : Identify suitable weighting operators that implement the criteria in Step 2, with examples given in Table 1. Compute QL using (14).

Step 4 – select parameters β , q : Compute the robustness margin (35) which dictates the amount of model uncertainty that can be tolerated, and also governs the effect of external disturbance on the tracking via (36). Tune parameter β (and q if used) based on knowledge of how well the model approximates the true dynamics. Larger values increase convergence speed, but reduce robustness.

Step 5 – fine-tuning: Apply ILC update (3) experimentally. If no evidence of instability is observed, increase parameters in Step 4 to improve convergence speed over all frequencies. Alternatively increase the magnitude of X over higher frequencies where convergence is desirable in Step 3, or increase the range of permissive inputs by increasing filter cut-off w_c . If evidence of instability is observed, make the opposite changes in Steps 3 and 4.

Implementation of the above procedure is now demonstrated through application of the framework to a realistic simulation scenario.

5. Application of framework

In order to demonstrate the effectiveness of this approach, control algorithms were implemented on a simulation of the Quanser QUBE servo rotational pendulum, carried out using MATLAB and Simulink. The full system model is given in Quanser (2014). This system is highly non-linear, open-loop unstable, and under-actuated, making it ideal for testing the proposed algorithm. A diagram of the system is shown in Figure 7. The input u to this system is the voltage V_m applied to the rotary arm, which actuates the arm angle q_1 and thus alters pendulum angle q_2 . These two angles combine to give the output, i.e. $y = [q_1, q_2]^\top$. All angles are in degrees.

To better replicate real-world conditions, band-limited white noise (noise power 5×10^{-7} , $SNR_{q_1} = 37.3$, $SNR_{q_2} = 12.6$) was added to the output angles, and the parameter values were altered slightly from the nominal values used to construct model G , as shown in Table 2. Initial conditions of $q_1 = 0$, $q_2 = 0.0873$, $\dot{q}_1 = 0.5$, $\dot{q}_2 = 0.5$ were selected to simulate a slight deviation from the equilibrium position. Note that the Quanser QUBE rotary encoder takes the initial angle q_1 to be zero regardless of absolute position, which is reflected in the choice of $q_1 = 0$.

The intended task was to stabilise the pendulum to the upright position whilst the rotary arm tracked a periodic

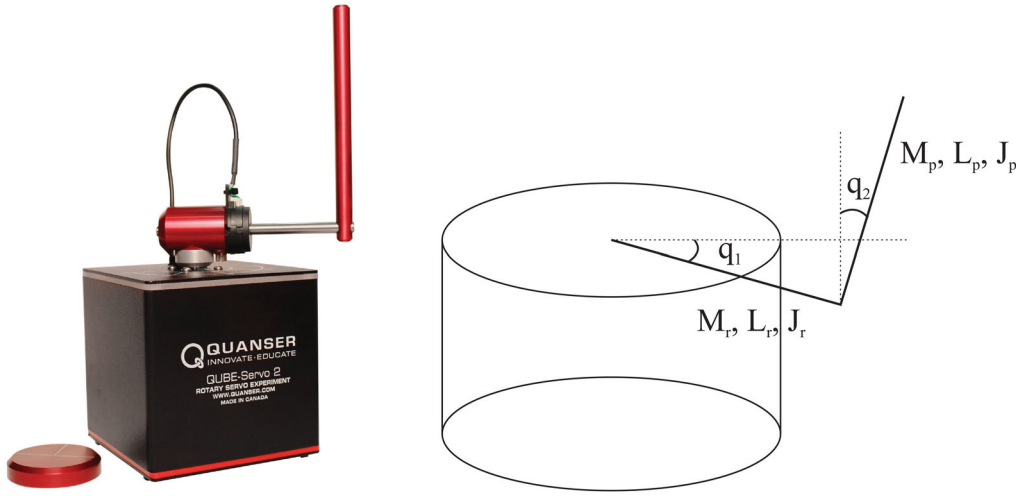


Figure 7. The rotary inverted pendulum system, with parameters given in Table 2.

Table 2. Pendulum nominal and model parameter values.

Parameter	Nominal	Model
M_p	Pendulum mass (kg)	0.024
M_r	Rotary arm mass (kg)	0.095
L_r	Rotary arm length (m)	0.085
L_p	Pendulum length (m)	0.129
J	Moment of inertia	$\frac{ML^2}{12}$
D_p	Pendulum damping coefficient	0.0012
R_m	Motor terminal resistance	8.4
k_m	Motor torque constant	0.042

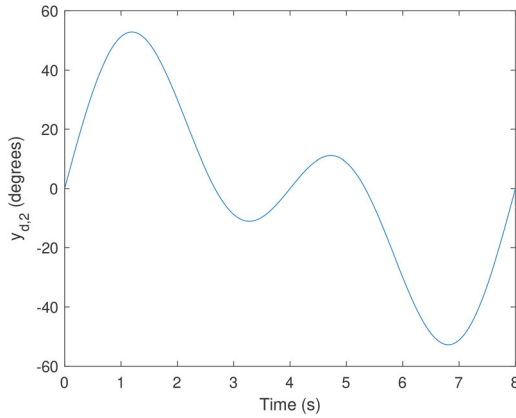


Figure 8. Servo angle component of reference signal y_d .

sinusoidal signal, as given by

$$y_d(t) = \left[30 \left(\sin\left(\frac{\pi t}{4}\right) + \sin\left(\frac{\pi t}{8}\right) \right), 0 \right]^T, \quad t \in [0, T] \quad (45)$$

where $T = 8$. This reference contains multiple low-frequency components, and is similar to that used in other ILC works, e.g. Zhang et al. (2022). The reference is shown in Figure 8.

A parallel control structure was used of the form of Figure 2(b), with two nested PID loops acting as the internal stabilising feedback controller K . This control structure allowed stabilisation of q_2 to be decoupled from tuning the performance of q_1 , simplifying the design procedure. The form of K is thus given by $K = (K_2 K_1, K_1)$ where K_1, K_2 are the PID control operators. These controllers were heuristically tuned in turn to achieve a stable closed-loop response, with resulting parameter values $K_{p,1} = 186, K_{i,1} = 542.7, K_{d,1} = 11.3, N_1 = 100, K_{p,2} = 0.0155, K_{i,2} = 0.00185, K_{d,2} = 0.0204, N_2 = 8.06$. The full system setup is shown in Figure 9.

The non-linear model G was then linearised about the upright position ($q_2 = 0$) and combined with the feedback controller K (according to the configuration shown in Figure 9) to generate the stable plant model P . A full summary of the linearisation procedure and construction of the system matrices is given in the supplementary material, available at <https://doi.org/10.5258/SOTON/D2914>.

5.1 ILC design

First the design procedure was implemented without the robustness filter ($Q = I$), in order to establish baseline performance. This corresponds to the choice $Z = 0$ within Step 3, which therefore means ILC will seek to achieve the minimum possible tracking error.

To satisfy Step 4, the choice $X = \beta(I + qPP^T)^{-1}$ is made, where $\beta \geq 0, q \geq 0$ are scalar. This allows for a transparent trade-off between convergence speed at different frequencies through parameter q . For large q it follows that X places emphasis on minimising the tracking error in the cost function (12) and also maximises convergence speed in (13) since $P^T X P = LP$ approximates

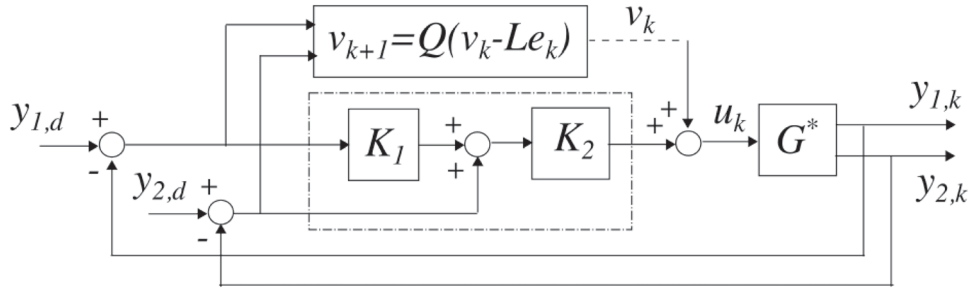


Figure 9. Parallel control implementation for the pendulum.

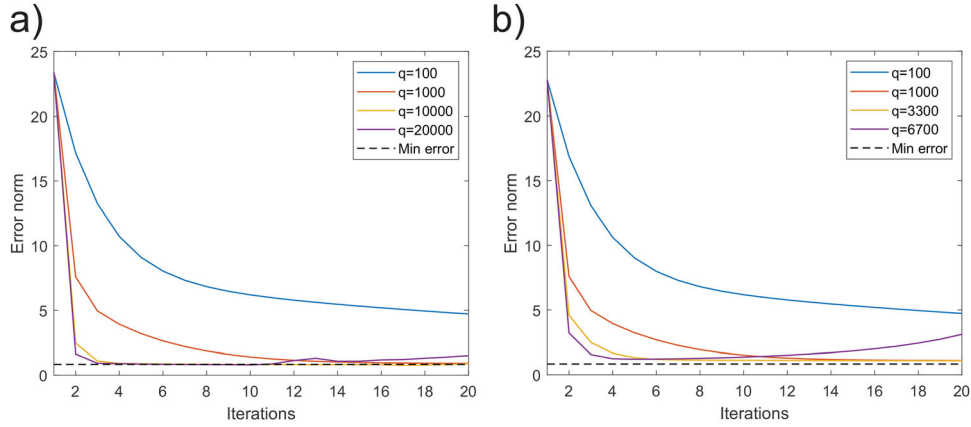


Figure 10. Convergence when using $X = q(I + qPP^T)^{-1}$, $Z = 0$ with varying $q \geq 0$ and (a) no uncertainty, (b) parametric model uncertainty (see Table 2).

unity and ζ approximates 0. Unfortunately the terms $\|LP\|$ and $\|L\|$ in robustness margin (34) are large, so that the system has a small robustness margin.

To address this, q can be decreased: X then tends towards βI and the convergence speed reduces while the robustness margin increases. The ILC algorithm produced using $X = \beta(I + qPP^T)^{-1}$ is the well-known norm-optimal ILC (Amann et al., 1996), which has proven extremely popular within the ILC community for many years. Furthermore, suitable choice of parameter q generates both inverse ($q \gg \frac{1}{\|PP^T\|}$) or gradient ($q = 0$) ILC.

Figure 10(a) compares the convergence performance for different q , with $\beta = q$ selected to allow convergence rate at lower reference frequencies to be directly tuned.

Increasing q improves the initial convergence, however, large values of q cause instability. This is exacerbated when implementing the algorithm on the system in the presence of model uncertainty, when the parameter values are altered to those shown in the right-hand column of Table 2. This is shown in Figure 10(a,b) and directly confirms the trade-off between robustness and convergence speed (i.e. minimising (13) and maximising (34)). Recall that larger values of q lead to $q \gg \frac{1}{\|PP^T\|}$, resulting in an algorithm that closely approximates inverse ILC (i.e.

the 2nd row of Table 1 with $\beta = 1$). As such, Figure 10 clearly demonstrates the improved convergence rate of this parameter choice, as well as the decreased robustness resulting from increased $\|LP\|$, $\|L\|$, ζ .

Figure 11 compares convergence when selecting $q = 0$, with different values of β (i.e. gradient ILC), in the presence of model uncertainty. Conclusions are the same as those from Figure 10, in that larger values of β (corresponding to larger $|X(\omega)|$ over all frequencies) lead

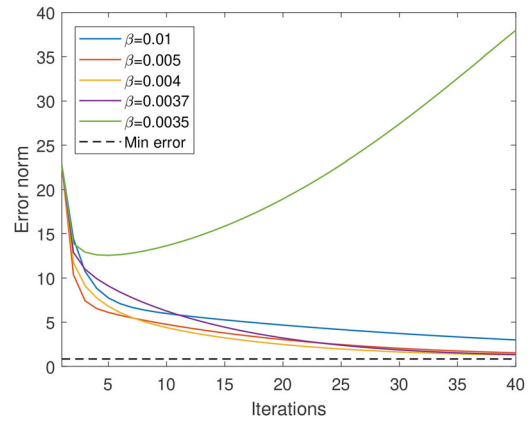


Figure 11. Convergence when using $X = \beta$, $Z = 0$ with parametric model uncertainty (see Table 2).

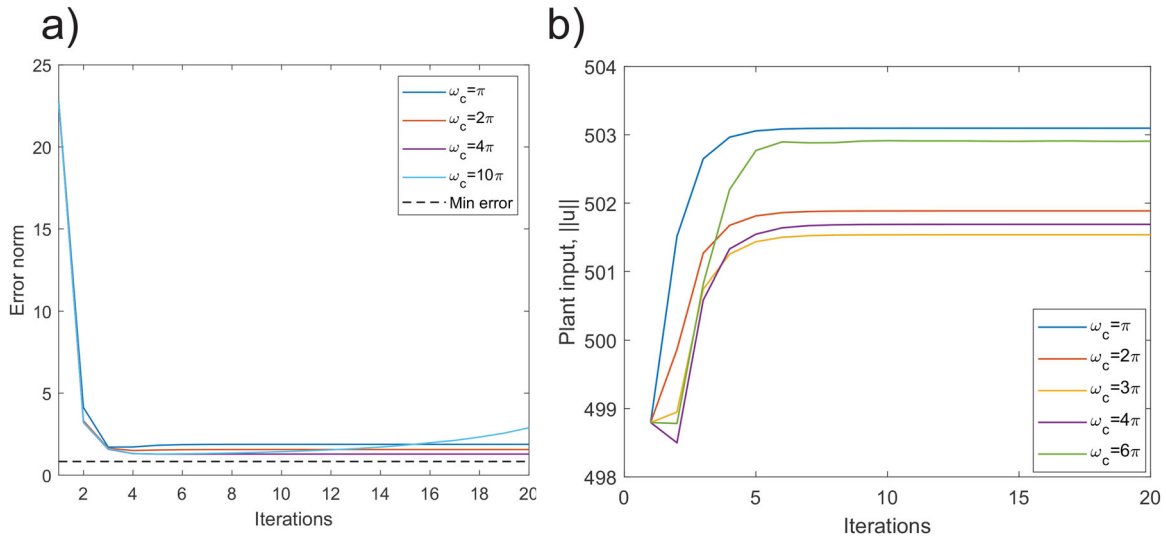


Figure 12. (a) error and (b) input convergence with different filter cutoff frequencies, using a value of $q = 6.7 \text{ e}^{+3}$ with added model uncertainty.

to instability. When comparing this performance to that of Figure 10, slower convergence is observed with $q = 0$, which backs up the theoretical conclusions of Section 2.2.

Having confirmed the design procedure for $Z = 0$, the use of $Z \neq 0$ to improve robust performance is now illustrated. In Step 3 of the design procedure a range of cut-off frequencies $\omega_c > \frac{\pi}{4}$ are selected, with Z being implemented as a high-pass zero-phase Butterworth filter in each case, consistent with the guidance given in the lower half of Table 1. This corresponds to Q as a high-pass filter, with a range of frequencies selected to allow the effectiveness of the approach to be examined. A filter order of 5 was selected to give a sharp cutoff.

Note that combining ILC update (3) and the expression for L in (14) results in

$$v_{k+1} = Qv_k + QLe_k = Qv_k + P^T X e_k. \quad (46)$$

This reveals that Q only operates on the previous control input, and hence filter Q can be implemented easily using the matlab 'filtfilt' command, or using an FFT filter. For operator X we use the same selection of $X = \beta(I + qPP^T)^{-1}$ as previously, with $\beta = q = 6.7 \text{ e}^{+3}$ corresponding to the case of instability for the uncertain system.

Figure 12(a) demonstrates the error convergence when using the ILC designs in the presence of parameter uncertainty. This reveals that lower cutoff frequencies (below 4Hz) are able to stabilise the system, but that these result in a greater steady-state error, as predicted by (25). Furthermore, Figure 12(b) demonstrates convergence of the input signal to a constant value, as predicted by Theorem 2.5.

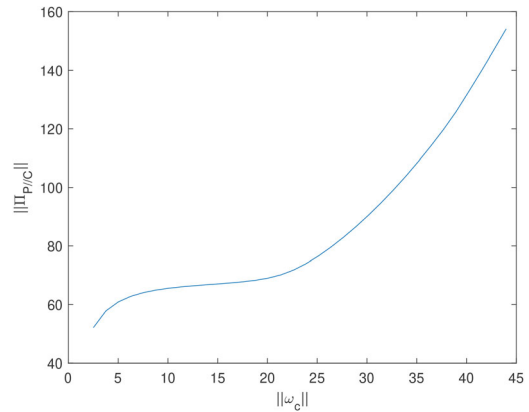


Figure 13. Change in gain bound inverse with altering cutoff frequency.

The increase in robustness observed when introducing the Q filter demonstrates the robustness results in Section 3. This can be seen in Figure 13, which shows how the inverse of the gain margin (i.e. the inverse of the right-hand side of (34)) increases with increasing cut-off frequency, corresponding to a smaller bound on the gap $\vec{\delta}(P, P^*)$ and hence decreased robustness to model uncertainty.

6. Conclusions and future work

This paper has introduced a comprehensive new approach to the design of ILC algorithms, allowing trade-off between error convergence and robustness in a systematic way. This has been achieved by minimising a cost function, and using the non-linear gap metric framework to analyse the resulting robustness properties. The

approach extended previous cost-function approaches through the inclusion of a feedback controller and a Q filter, and the introduction of a transparent, unified design procedure to implement the framework.

The theory was demonstrated by simulation on an inverted pendulum, revealing an increase in performance and decrease in robustness when increasing the learning matrix L , with the reverse being true for the robustness filter Q . These findings align with both the standard ILC literature and the theory presented in this paper. Having validated the framework in a controlled numerical scenario where all aspects can be defined and quantified, the next step will focus on implementing it on an experimental system to demonstrate its practicality and effectiveness in the real-world.

Whilst the new design procedure involves tuning the matrices L and Q sequentially, the framework could be expanded to directly use (12) in a quantitative manner. Quantifying robustness and performance in terms of the two ILC operators opens up the possibility of tuning L and Q such that robustness is optimised given certain performance criteria, or vice-versa.

Longer term future work areas include relaxing the assumptions of a fixed reference trajectory. Steps have already been made in this direction in the form of point-to-point ILC and spatial ILC, which relax the tracking objective. These can be extended and placed in the robustness framework introduced in this paper. Greater flexibility in the task then allows the range of practical applications of ILC to be extended. A prominent example is ILC-enabled stroke rehabilitation, where robotics or electrical stimulation are controlled to assist human motion.

Note

1. The biased norm is defined as $\|w\|_{\bar{w}} = \|w - \bar{w}\|$ for signal w and bias $\bar{w}$

Acknowledgments

The authors would like to thank Christian Webb for his assistance with the writing of this paper.

Data availability statement

All data supporting this study are openly available from the University of Southampton repository at <https://doi.org/10.5258/SOTON/D2914>.

Disclosure statement

The authors have no relevant financial or non-financial interests to disclose.

References

- Ahn, H., Moore, K. L., & Chen, Y. (2005). Schur stability radius bounds for robust iterative learning controller design. In *Proceedings of the 2005 American Control Conference* (Vol. 1, pp. 178–183). IEEE.
- Amann, N., Owens, D. H., & Rogers, E. (1996). Iterative learning control for discrete-time systems with exponential rate of convergence. *IEEE Proceedings – Control Theory and Applications*, 143(2), 217–224. <https://doi.org/10.1049/ip-cta:19960244>
- Arimoto, S., Kawamura, S., & Miyazaki, F. (1984). Bettering operation of robots by learning. *Journal of Robotic Systems*, 1(2), 123–140. <https://doi.org/10.1002/rob.4620010203>
- Binz, R., & Aronovskiy, S. (2021). Iterative learning control strategy for a furuta pendulum system with variable-order linearization. *IFAC-PapersOnLine*, 54(20), 14–19. <https://doi.org/10.1016/j.ifacol.2021.11.146>
- Bradley, R. (2010). *The robust stability of iterative learning control* [Unpublished doctoral dissertation]. University of Southampton.
- Bristow, D. A. (2008). Weighting matrix design for robust monotonic convergence in norm optimal iterative learning control. In *2008 American Control Conference* (pp. 4554–4560). IEEE.
- Bristow, D., Tharayil, M., & Alleyne, A. (2006). A survey of iterative learning control. *IEEE Control Systems Magazine*, 26(3), 96–114. <https://doi.org/10.1109/MCS.2006.1636313>
- Chen, S. (2022). *Decentralised iterative learning control for high performance collaborative tracking and formation objectives* [Unpublished doctoral dissertation]. University of Southampton.
- Chen, S., & Freeman, C. T. (2019). Decentralised collaborative iterative learning control for MIMO multi-agent systems. In *2019 American Control Conference (ACC)* (pp. 3352–3357). IEEE.
- Chi, R., Hou, Z., Jin, S., & Wang, D. (2013). A data-driven optimal design of point-to-point ILC. In *Proceedings of the 32nd Chinese Control Conference* (pp. 2934–2938). IEEE.
- Donkers, T., van de Wijdeven, J., & Bosgra, O. (2008). Robustness against model uncertainties of norm optimal iterative learning control. In *2008 American Control Conference* (pp. 4561–4566). IEEE.
- Freeman, C. T. (2017). Robust ILC design with application to stroke rehabilitation. *Automatica*, 81, 270–278. <https://doi.org/10.1016/j.automatica.2017.04.016>
- Freeman, C. T., Lewin, P. L., & Rogers, E. (2009). Discrete Fourier transform based iterative learning control design for linear plants with experimental verification. *Journal of Dynamic Systems, Measurement and Control*, 131(3), 1–10.
- Georgiou, T. T., & Smith, M. C. (1997). Robustness analysis of nonlinear feedback systems: An input–output approach. *IEEE Transactions on Automatic Control*, 42(9), 1200–1221. <https://doi.org/10.1109/9.623082>
- Hao, S., Liu, T., & Gao, F. (2019). PI based indirect-type iterative learning control for batch processes with time-varying uncertainties: A 2D FM model based approach. *Journal of Process Control*, 78, 57–67. <https://doi.org/10.1016/j.jprocont.2019.04.003>
- Hazarika, H., & Swarup, A. (2020). Application of an optimal ILC algorithm for flow rate tracking of a ventilator system. In *2020 1st IEEE International Conference on Measurement*,

- Instrumentation, Control and Automation (ICMICA)* (pp. 1–6). IEEE.
- Meindl, M., Campe, F., Lehmann, D., & Seel, T. (2024). Solving motion tasks with challenging dynamics by combining kinodynamic motion planning and iterative learning control. In *2024 European Control Conference (ECC)* (pp. 1208–1213). IEEE.
- Owens, D. H. (2016). *Iterative learning control – an optimization paradigm*. Springer.
- Owens, D. H., Freeman, C. T., & Chu, B. (2014). An inverse-model approach to multivariable norm optimal iterative learning control with auxiliary optimisation. *International Journal of Control*, 87(8), 1646–1671. <https://doi.org/10.1080/00207179.2014.880951>
- Precup, R. E., Gavriluta, C., Radac, M. B., Preitl, S., Dragos, C. A., Tar, J., & Petriu, E. (2009). Iterative learning control experimental results for inverted pendulum crane mode control. In *2009 IEEE International Conference on Systems, Man and Cybernetics* (pp. 323–328). IEEE.
- Qian, K., Li, Z., Asker, A., Zhang, Z., & Xie, S. (2021). Robust iterative learning control for pneumatic muscle with state constraint and model uncertainty. In *Proceedings IEEE International Conference on Robotics and Automation* (pp. 5980–5987). IEEE.
- Qiu, L., & Zhao, D. (2021). Robust control in gap metric. In J. Baillieul, & T. Samad (Eds.), *Encyclopedia of systems and control* (pp. 1941–1947). Springer International Publishing.
- Quanser (2014). *QUBE servo state space workbook*. <https://www.quanser.com/courseware-resources>
- Xu, Q., Li, Y., Cheng, J., & Xiao, T. (2020). Discrete-time adaptive ILC for uncertain systems with unknown nonlinear dead zone inputs and control directions. In *2020 IEEE 9th Data Driven Control and Learning Systems conference (DDCLS)* (pp. 923–928). IEEE.
- Xu, J., & Xu, J. X. (2013). Iterative learning control for output-constrained systems with both parametric and non-parametric uncertainties. *Automatica*, 49(8), 2508–2516. <https://doi.org/10.1016/j.automatica.2013.04.039>
- Yu, Q., Hou, Z., & Xu, J. X. (2018). D-type ILC based dynamic modeling and norm optimal ILC for high-speed trains. *IEEE Transactions on Control Systems Technology*, 26(2), 652–663. <https://doi.org/10.1109/TCST.2017.2692730>
- Zhan, X. (2010). Robust iterative learning control for nonlinear system based on T-S model. In *2010 Chinese Control and Decision Conference* (pp. 2115–2119). IEEE.
- Zhang, Y., Chu, B., & Shu, Z. (2022). Model-free predictive optimal iterative learning control using reinforcement learning. In *2022 American Control Conference (ACC)* (pp. 3279–3284). IEEE.