

An alternative Update Formula for Nonlinear Model-Based Iterative Learning Control

J.J.A. Eksteen^{a*} and P.S. Heyns^a

^a*Department of Mechanical and Aeronautical Engineering, University of Pretoria*

(Received 00 Month 20XX; final version received 00 Month 20XX)

The conventional Iterative Learning Control (ILC) algorithm for model-based ILC of nonlinear systems is presented with use of a nonlinear inverse model as ILC compensator. The nonlinear inverse model is solved with stable inversion. In addition an alternative ILC algorithm for model-based ILC of nonlinear systems is developed, also with using a nonlinear inverse model as ILC compensator. Some connections between the conventional and alternative ILC algorithms and Picard, Mann and Ishikawa iteration are explored. In a number of theoretical examples the conventional and alternative algorithms are compared.

Keywords: Iterative learning control, Stable inversion, Picard iteration, Mann iteration, response reconstruction, fatigue testing, discrete time, nonlinear.

1. Introduction

Consider a closed-loop control system represented by an operator T , which we would like to track the desired plant output signal y_d as closely as possible. Iterative learning control (ILC) is a repetitive control scheme that uses a learning capability to improve the tracking accuracy over repeated test trials of the closed-loop control system [1]. The learning mechanism involves calculating the tracking error of the previous iteration, and then calculating a corrective action from the tracking error and an inverse model of T , which is added to the previous input signal to obtain an “updated” input signal. The updated input signal is intended to improve the tracking accuracy of the next test trial, following which the tracking error may be calculated again and the process repeated. Under sufficient conditions for convergence of the algorithm the desired output can be exactly tracked, possibly over a reduced test bandwidth as may be required to ensure convergence.

ILC is often presented in the context of systems that have to repeatedly track the same desired output signal, for example industrial robots working on serial production lines that repeatedly do the same welding or spray painting job. In this case the real-time control system is typically some type of asymptotic tracking control system, and ILC then uses the results of the repeated tracking exercises to further improve the tracking accuracy (for early references see, e.g., [2] and [3]). This is done on an ongoing basis, and therefore ILC not only optimizes tracking accuracy for a given desired output and one specific set of system dynamics, but can also be used to *maintain* tracking accuracy by responding to gradual changes in the system dynamics through its “learning” capability.

When ILC is implemented purely to obtain the system input resulting in the exact

*Corresponding author. Email: jan.eksteen@up.ac.za

tracking of the desired output for a given set of system dynamics and then terminated once this has been achieved, it implies ILC is being used merely to once-off *reconstruct* a desired system response (i.e. output) rather than to *maintain* tracking accuracy over repeated test trials. When used in this way the ILC procedure is referred to here as *response reconstruction*.

We can also replace the physical test system with a system model and perform the test trials “mathematically”, in which case ILC is being used as an inversion procedure for the system model. The fact that ILC is capable of exact tracking (subject to sufficient conditions) implies that the inversion of the model will be exact. It is noteworthy that when we use ILC in this way to invert a model, we do not calculate the inverse model, but only use the model in the normal sense as if it is a physical test system. Using ILC in this way is an *indirect* way of inverting a model because the exact inverse system model is never calculated, as opposed to the *direct* approach followed in stable inversion. The use of ILC to accomplish inversion of nonlinear models was first proposed in [4] to the authors’ knowledge.

While ILC is capable of achieving convergence with even very simple forms of the ILC compensator L , such as a constant gains matrix (the so-called P-type ILC), the rate of convergence achieved with these compensators is not necessarily monotone. On the contrary, the convergence error (and achieved test outputs) may grow very large before finally decaying to zero. Achieving monotone convergence over relatively wide frequency bands generally requires the use of an approximate system inverse model in L . This is referred to here as *inverse model-based ILC*. The rate of convergence may, additionally, be slowed down by scaling down L using a scalar scale factor. These aspects have already been explored in ILC; see, for example, [5].

While ILC is capable of achieving convergence with even very simple forms of the ILC compensator L , such as a constant gains matrix (the so-called P-type ILC), the rate of convergence achieved with these compensators is not necessarily monotone. On the contrary, the convergence error (and achieved test outputs) may grow very large before finally decaying to zero. Achieving monotone convergence over relatively wide frequency bands generally requires the use of an approximate system inverse model in L . This is referred to as *inverse model-based ILC*. The rate of convergence may, additionally, be slowed down by scaling down L using a scalar scale factor. These aspects have already been explored in ILC; see, for example, [5].

When using ILC to invert a nonlinear system model, an approximate linear inverse model may be calculated for use in the ILC compensator. Such a model may be obtained by linear system identification, and as such represents a linearized version of the nonlinear test system around the average operating point represented by the identification data. The linear inverse model may be improved as an approximation of the actual nonlinear inverse model by generating it as the inverse of a linearization of the nonlinear model around the average operating point represented by the input signal of the previous ILC iteration. The linearization and inverse may furthermore be recalculated at every time point for the state of the input at that time (rendering it time varying). Modifying the ILC-based inversion of a nonlinear model in this way results in the Newton method of inverting a system (with appropriate reformulation of the system), which has recently been presented by [7] for continuous-time systems. For a brief introduction to the Newton method of solving nonlinear equations, see [8]. This shows the ILC-based inversion of a nonlinear model using an approximate linear time-invariant (LTI) model to be a special case of the novel Newton inversion method. As a purely ILC approach to be executed on physical test systems this approach is referred to as Newton method based ILC, and was first presented for nonlinear discrete time systems by [9]. Promising to realize the advantages of rapid convergence of the Newton method in ILC, it is worthy of future

investigation as an alternative to more complex methods utilizing a nonlinear inverse model in the ILC compensator (discussed next). [9] also examine convenient ways of numerically implementing the algorithm, as does [7].

In this study as a novel approach we implement *nonlinear*, inverse model-based ILC compensators, that are solved directly by stable inversion [10], [11], in ILC of nonlinear test systems, as opposed to using linear inverse-based compensators. The hope is that in case of divergence of ILC with a linear L the use of a nonlinear (and potentially more accurate) L may result in convergence, may widen the frequency range over which convergence occurs, or may increase the accuracy of the best results achieved prior to divergence. We will also show how an accurate (nonlinear) inverse model in L simplifies the ILC dynamics (also shown in [4]), resulting in monotone convergence and allowing better control of the rate of convergence.

We also develop an altogether new ILC algorithm, and show that the alternative algorithm demonstrates the same properties as the conventional algorithm with regard to convergence and control of the rate of convergence. The alternative algorithm is relevant only when using a nonlinear L since, when using a linear L the alternative algorithm is equivalent to the conventional ILC algorithm.

Finally, we also note that the use of Mann [12] and Ishikawa iteration in stable inversion [13] has an analogy in ILC in that the conventional and alternative ILC algorithms developed here both have parallels in the Picard and Mann iteration schemes. It is furthermore shown that the application of Ishikawa iteration to ILC result in novel ILC iteration schemes for both the conventional and alternative ILC algorithms.

In Section 3, the conventional ILC algorithm is presented for model-based or inverse model-based ILC on *nonlinear* systems for the deterministic, square, MIMO case. A *general inverse based* (GIB) ILC compensator, which explicitly uses the approximate system inverse model, is proposed, and represents a slight modification of existing approaches for ILC in robotics literature in that it includes a scalar scale factor for controlling the rate of convergence. It is demonstrated in Section 3.5 how ILC with the GIB compensator employing the scale factor represents a Mann iteration version of ILC without the scale factor (that corresponds to Picard iteration).

The alternative ILC algorithm is presented in Section 4, as well as a modified version that enables it to fully match the various properties of the conventional algorithm. Theorems 3.4 to 3.9 are modifications for the alternative and modified alternative ILC algorithms of theorems in the literature of the conventional ILC algorithm.

In Example 1 we demonstrate the convergence of the conventional and alternative ILC algorithms for a short-duration deterministic signal, and in Example 2 we present an example where the alternative ILC algorithm converges while the conventional algorithm diverges for a random signal. In Example 3 we perform ILC using the inverse of a linear approximate model. In Example 4 we perform ILC using the inverse of a nonlinear approximate model, and compare the results with those of Example 3. Both the linear and nonlinear models of Examples 3 and 4 are obtained from system identification. In each case we will also compare the results of ILC using the conventional and alternative ILC algorithms.

2. Model-Based ILC Compensators

Achieving monotone convergence generally requires the use of model-based ILC compensators, with a number of types in existence that are aimed at this need, including the following designs:

- Contraction Mapping ILC Compensator [14], [15]

- Phase-Cancellation ILC Compensator [16]
- The Modified Phase-Cancellation ILC Compensator [17]
- The α Pseudoinverse-Based ILC Compensator [18].
- Inverse ILC Compensators, (see e.g. [19] and [20]).

The inverse ILC Compensator is also the form of the ILC compensator in the earliest implementations of automotive service load simulation for fatigue testing [21]. Early implementations employed frequency domain inverses. In case of non-minimum phase systems, when the time-domain inverses have unstable dynamics, the approach was to approximate the true inverse with an approximate causal inverse. This limitation can be removed by using the non-causal stable inversion method for non-minimum phase systems. This allows the most accurate inverse that can be identified for the test system to be used in ILC. The advantages of using an accurate inverse as ILC compensator are the following:

- Widest possible frequency band of convergence;
- Monotonic decay of the error of the input signal with respect to the limit input signal;
- Potentially rapid convergence (if desired); and
- Good control over the rate of the convergence of the input signal (this may be achieved using a scale factor in the inverse compensator).

In practice the model uncertainty usually increases with frequency due to noise and/or nonlinear effects manifesting at high frequencies in conjunction with the natural attenuation of typical systems at high frequencies, all of which reduces accuracy of identified models at high frequencies. The model uncertainty at high frequencies in combination with the high gain of the inverse compensator at high frequencies can limit the bandwidth over which convergence can be achieved. A second disadvantage of the inverse compensator is that it is limited to square systems, unlike the pseudo-inverse compensator. The α pseudoinverse compensator is in fact a generalization of the inverse compensator for non-square systems. In practice, however, square systems are common.

The problem of limited convergence bandwidth of the inverse-based compensator may be alleviated by opting for one of the compensators with reduced gain at high frequencies (phase cancellation or α -pseudoinverse compensators), or by incorporating a general zero-phase filter in the compensator to attenuate the high frequency response. Opting for this approach leads to a general, inverse-based compensator, given in operator form as

$$L = C\tilde{L},$$

with C a zero-phase filter and \tilde{L} the approximate inverse of T . A similar approach may be found in [22] and [23]. For specific choices of C the compensator includes all the different choices of model-based compensators described above. By the proper design of C the compensator allows the usually high magnitude of \tilde{L} at the high frequencies to be attenuated for the sake of robustness of convergence against model error. To illustrate, with this compensator the frequency domain version of the condition for monotone convergence of ILC of linear systems yields

$$\| |Q_0(e^{j\omega})|^2(1 - |C_0(e^{j\omega})|^2\tilde{L}(e^{j\omega})T(e^{j\omega})) \|_\infty < 1.$$

At frequencies where the product of $\tilde{L}(e^{j\omega})T(e^{j\omega})$ would normally violate the condition (due to model error in \tilde{L}), C may be used to attenuate the product at the relevant frequencies in order to satisfy the inequality and thus retain convergence at that frequency.

3. Inverse Model Based ILC of Nonlinear Systems

3.1. Test Formulation

Consider a $m \times m$ discrete time, nonlinear, stable, injective test system, represented by a nonlinear operator T with $T([0]) = [0]$. The input signal for the i -th test trial is designated $u^{(i)}$ and the output signal $y^{(i)}$, so that (with slight abuse of notation)

$$y^{(i)} = T(u^{(i)}) . \quad (1)$$

The system is relaxed at the beginning of every test trial. We assume there exists a desired plant output $y_d(k)$, $k = 0, \dots, N - 1$, $y_d(0) = y_d(N) = [0]$, which we designate y_d , and which we desire to track as closely as possible. By the injectiveness of the plant there exists a unique $u_d(k)$, $k = 0, \dots, N - 1$, designated u_d , such that [11]

$$y_d = T(u_d) . \quad (2)$$

3.2. Algorithm

The update formula in the conventional algorithm for ILC on both linear and nonlinear systems is frequently given as ([23] and [4])

$$u^{(i+1)} = Q(u^{(i)} + L(e^{(i)})) , \quad (3)$$

with

$$e^{(i)} = y_d - y^{(i)} , \quad (4)$$

$y^{(0)} = 0$, and Q a zero-phase, discrete-time linear filter. While usually linear, in this study the ILC compensator L is a generally nonlinear, discrete-time transfer operator.

When T is nonlinear, convergence to the desired u_d can potentially be achieved with Eq. 3 even when L is the inverse of an approximate model of the system. However, the convergence generally can not be achieved in one step, irrespective of how accurate a representation of the true inverse of T the compensator L is. This can be rectified by slightly modifying Eq. 3 as follows ([4] and [7]):

$$u^{(i+1)} = Q(u^{(i)} + L(y_d) - L(y^{(i)})) . \quad (5)$$

This algorithm is shown in Fig. 1. In case of a nonlinear T this update formula is theoretically capable of achieving convergence in one step, but that requires that $L = T^{-1}$. Note that for linear L Eq. 5 reduces to the more conventional form (Eq. 3). In the sequel we will use the modified update formula, Eq. 5, as the standard representation of the conventional approach to ILC with nonlinear test systems, together with Eq. 1 and Eq. 2. In view of the alternative algorithm proposed in the next section we will refer to it as the *conventional* algorithm.

3.3. Convergence

Next we focus on the convergence of the conventional algorithm. By inserting Eq. 1 into Eq. 5 the system formulation of Eq. 5 in the iteration domain is obtained as

$$u^{(i+1)} = Q(u^{(i)} + L(y_d) - L(T(u^{(i)}))) \quad (6)$$

$$:= \mathcal{T}(u^{(i)}, y_d). \quad (7)$$

Eq. 6 is a fixed point problem in the variable u and we may follow the standard route of using the contraction mapping theorem in terms of the nonlinear operator \mathcal{T} to obtain a strong condition for both the existence and uniqueness of a solution. [4] provides a more relaxed condition for convergence, at the expense of having to prove the iteration is bounded and the existence of a well-defined limit signal separately. Here we follow the approach of [4], starting with the boundedness of the ILC algorithm.

Theorem 3.1: If T is bounded input-bounded output (BIBO) stable in the 2-norm and $\alpha\beta < 1$ with $\alpha = \|Q\|_\infty$ and

$$\beta = \sup_{u \neq 0} \frac{\|u - L(T(u))\|_2}{\|u\|_2}, \quad (8)$$

then Eq. 5 is bounded input-bounded output (BIBO) stable in the 2-norm. For the proof, see [4]. \diamond

Noting that

$$u^{(\infty)} = \lim_{i \rightarrow \infty} u^{(i)} = \lim_{i \rightarrow \infty} u^{(i+1)}, \quad (9)$$

the limit signals for Eq. 5 and Eq. 6 are defined by

$$u^{(\infty)} = Q(u^{(\infty)} + L(y_d) - L(y^{(\infty)})) \quad (10)$$

$$= Q(u^{(\infty)} + L(y_d) - L(T(u^{(\infty)}))) \quad (11)$$

If the limit signals exist, then for $Q = 1$ clearly $L(y_d) = L(y^{(\infty)})$, implying $y^{(\infty)} = y_d$ since L is injective, and in turn $u^{(\infty)} = T^{-1}(y^{(\infty)}) = T^{-1}(y_d) = u_d$ by the injectiveness of T . If $Q \neq 1$, then (from Eq. 10)

$$u^{(\infty)} = Q(1 - Q)^{-1}(L(y_d) - L(y^{(\infty)})) \quad (12)$$

and $y^{(\infty)} = T(u^{(\infty)})$. We have the following theorem for existence of the fixed points:

Theorem 3.2: If the conditions of Theorem 3.1 hold and L is BIBO stable, then for $Q \neq 1$ Eq. 12 is BIBO stable in the 2-norm and $u^{(\infty)}$ and $y^{(\infty)}$ are well defined. If $Q = 1$ then $u^{(\infty)}$ and $y^{(\infty)}$ are well defined, with $y^{(\infty)} = y_d$ and $u^{(\infty)} = u_d$. The proof for the case $Q = 1$ follows from the preceding discussion. For the proof of the case $Q \neq 1$, see [4]. \diamond

The iteration domain formulation for the convergence error of the input signal w.r.t.

the limit signal is then (using Eq. 6 and Eq. 11)

$$\Delta^{(i+1)} = u^{(\infty)} - u^{(i+1)} \tag{13}$$

$$\begin{aligned} &= Q(u^{(\infty)} + L(y_d) - L(T(u^{(\infty)}))) - Q(u^{(i)} + L(y_d) - L(T(u^{(i)}))) \\ &= Q(u^{(\infty)} - u^{(i)} - (L(T(u^{(\infty)})) - L(T(u^{(i)})))) \\ &= Q\Delta^{(i)} - Q(L(T(u^{(\infty)})) - L(T(u^{(\infty)} - \Delta^{(i)}))) \end{aligned} \tag{14}$$

$$= Q\Delta^{(i)} - Q\Phi(u^{(\infty)}, \Delta^{(i)}) , \tag{15}$$

with

$$\Phi(u^{(\infty)}, \Delta^{(i)}) := L(T(u^{(\infty)})) - L(T(u^{(\infty)} - \Delta^{(i)})) . \tag{16}$$

We have the following sufficient condition for convergence:

Theorem 3.3: If the ILC update formula Eq. 5 is BIBO stable, the limit signal of Eq. 12 is well defined, and $\alpha\gamma < 1$ with $\alpha = \|Q\|_\infty$ and

$$\gamma = \sup_{\Delta \neq 0} \frac{\|\Delta - \Phi(u^{(\infty)}, \Delta)\|_2}{\|\Delta\|_2} , \tag{17}$$

then $\|\Delta^{(i+1)}\| < \alpha\gamma\|\Delta^{(i)}\|$, that is, the input convergence error is monotonically decreasing and the input $u^{(i)}$ converges to the input limit signal. For the proof, see [4].
 \diamond

3.4. Choice of ILC Compensator

With regard to model-based designs of L , in case of square test systems L may be the (linear) inverse of the linear model that approximates the nonlinear system [5], or in case of non-square systems it may be the α pseudo-inverse [6] of the linear approximation. The inverse compensator we employ here is given in operator form as

$$L = cC\tilde{L} , \tag{18}$$

with c a real scalar, C a linear, zero-phase filter, and \tilde{L} the linear or nonlinear approximate inverse of the generally nonlinear system T . This compensator is referred to here as the *general inverse-based* (GIB) compensator. In this study the focus is on using both linear and nonlinear inverse models in \tilde{L} . Such models may be obtained by system identification on behavioral data of the nonlinear system T , and inverting the resulting model. The resulting inverse is often solved in the frequency domain in case of linear models, but may also be solved in the time domain using stable inversion in case of both linear and nonlinear models to circumvent the instability of the inverse associated with nonminimum phase zeros of the normal model.

Eq. 3 used in conjunction with the GIB compensator (Eq. 18) is also the form of the ILC algorithm in response reconstruction for purposes of automotive fatigue testing. For implementations with a linear inverse in \tilde{L} see [21], [24], [25], [26], [27], and [28]. For an implementation with a nonlinear inverse in \tilde{L} see [7]. For an aeronautical fatigue testing application using a linear inverse in \tilde{L} , see [29] (ignore comments therein about

the relative accuracy of the time domain vs. the frequency domain system identification approaches).

Substitution of the GIB compensator (Eq. 18) and Eq. 16 into Eq. 17 gives

$$\begin{aligned}\gamma &= \sup_{\Delta \neq 0} \frac{\|\Delta - \Phi(u^{(\infty)}, \Delta)\|_2}{\|\Delta\|_2} \\ &= \sup_{\Delta \neq 0} \frac{\|\Delta - (L(T(u^{(\infty)})) - L(T(\Delta + u^{(\infty)})))\|_2}{\|\Delta\|_2}\end{aligned}\quad (19)$$

$$= \sup_{\Delta \neq 0} \frac{\|\Delta - cC(\tilde{L}(T(u^{(\infty)})) - \tilde{L}(T(\Delta + u^{(\infty)})))\|_2}{\|\Delta\|_2}\quad (20)$$

When the convergence condition of Theorem 3.3, namely $\alpha\gamma < 1$ is violated, the role of a small c in recovering convergence (at the expense of a slower rate of convergence) is clear.

For the ideal GIB compensator with $\tilde{L} = T^{-1}$ and $C = 1$, i.e. $L = cT^{-1}$, Eq. 14 becomes

$$\begin{aligned}\Delta^{(i+1)} &= Q\Delta^{(i)} - Q(cT^{-1}(T(u^{(\infty)})) - cT^{-1}(T(u^{(\infty)} - \Delta^{(i)}))) \\ &= (1 - c)Q\Delta^{(i)}.\end{aligned}\quad (21)$$

Clearly, if $c = 1$, $\Delta^{(i+1)} = [0]$, i.e. convergence is achieved in one iteration, with the limit signals as given in Theorem 3.2. This results confirms the advantages of using an accurate inverse model in the ILC compensator: rapid convergence, widest possible frequency band of convergence (up to Nyquist frequency), precise control over the rate of convergence of $u^{(i)}$, and monotonic convergence of $u^{(i)}$.

3.5. Connections with Fixed Point Iteration Methods

The system formulation in the iteration domain for the GIB compensator (Eq. 18) with $c = 1$ and $Q = 1$ becomes (from Eq. 6)

$$u^{(i+1)} = u^{(i)} + cC\tilde{L}(y_d) - cC\tilde{L}(T(u^{(i)})),\quad (22)$$

$$= u^{(i)} + C\tilde{L}(y_d) - C\tilde{L}(T(u^{(i)}))\quad (23)$$

$$:= \mathcal{T}_{11}(u^{(i)}, y_d).\quad (24)$$

Here we view $\mathcal{T}_{11}(u^{(i)}, y_d)$ as a type of baseline approach for ILC with the GIB compensator (with $c = 1$) and that corresponds to the Picard fixed point iteration approach. If we now formulate a standard Mann iteration scheme with the operator $\mathcal{T}_{11}(u^{(i)}, y_d)$, the result is

$$u^{(i+1)} = (1 - \alpha_i)u^{(i)} + \alpha_i\mathcal{T}_{11}(u^{(i)}, y_d)\quad (25)$$

$$= u^{(i)} + \alpha_i C\tilde{L}(y_d) - \alpha_i C\tilde{L}(T(u^{(i)}))\quad (26)$$

with $\alpha_i \in (0, 1]$. Comparing Eq. 26 with Eq. 22 shows that it resembles the system formulation for ILC with $Q = 1$ and with the GIB compensator with $c = \alpha_i$. Thus,

performing ILC with the conventional algorithm with $Q = 1$ and the GIB compensator with $c \in (0, 1)$ essentially represents a special case of Mann iteration (using a constant α) with the operator $\mathcal{T}_{11}(u^{(i)}, y_d)$. This will be true for any ILC compensator featuring a constant scale factor. The use of the gain $c \in (0, 1)$ in such compensators thus has the same advantages for achieving convergence compared to the $c = 1$ case as Mann iteration (with a constant α) has compared to Picard iteration.

Note that it is possible to form an Ishikawa iteration scheme with $\mathcal{T}_{11}(u^{(i)}, y_d)$ as follows:

$$\begin{aligned} u^{(i+1)} &= (1 - \alpha_i)u^{(i)} + \alpha_i \mathcal{T}_{11}(u^{(i)}, y_d) \\ \mu^{(i)} &= (1 - \beta_i)u^{(i)} + \beta_i \mathcal{T}_{11}(u^{(i)}, y_d) , \end{aligned} \quad (27)$$

with $\alpha_i \in (0, 1]$ and $\beta_i \in [0, 1]$. It can be shown that a fixed point of Picard iteration is also a fixed point of Mann and Ishikawa iteration. Investigating the connections (if any) between Ishikawa iteration and existing ILC iteration schemes is a subject of future research.

4. Alternative ILC Algorithm Using a Nonlinear Inverse Model

4.1. Development

We consider the same nonlinear system and associated assumptions as in Section 3. Towards deriving a fundamentally different form of the ILC algorithm, we retain the Q filter in an optional capacity, but for the moment consider the $Q = 1$. When the plant output converges to y_d , the plant input at the same time converges to u_d . If the plant input, instead of being the parameter that is updated, is instead directly obtained as output of the ILC compensator L , then as the plant input converges to u_d , the input to the compensator, say \tilde{y} , will converge to a limit value \tilde{y}_d so that

$$u_d = L(\tilde{y}_d) . \quad (28)$$

If we can formulate an update formula in the \tilde{y} parameter, then the goal will be for \tilde{y} to converge to \tilde{y}_d , because that will imply desired convergence of u and y to u_d and y_d respectively. This suggests the possibility of an update formula on the output side of the plant instead of on the input side as with conventional ILC (keep in mind L is essentially an inverse model of the plant). The following algorithm follows this approach. As before

$$y^{(i)} = T(u^{(i)}) . \quad (29)$$

We define a new parameter as input to the ILC compensator which, being an input to the ILC compensator, is related in type to the system output and thus we use the symbol \tilde{y} , and have

$$u^{(i)} = L(\tilde{y}^{(i)}) . \quad (30)$$

The new update formula is in the \tilde{y} parameter, and is as follows:

$$\tilde{y}^{(i+1)} = Q(\tilde{y}^{(i)} + y_d - y^{(i)}) , \quad (31)$$

with initial value $\tilde{y}^{(0)} = 0$. If $Q = 1$ in Eq 31 then clearly, when $y^{(i)}$ converges to y_d , $\tilde{y}^{(i)}$ converges to a limit value that, working back through Eq. 29 and Eq. 30 for $y^{(i)} = y_d$

must be \tilde{y}_d as defined in Eq. 28. In other words, when $\tilde{y}^{(i)}$ converges, by definition u and y converges to u_d and y_d respectively, which is the goal of ILC. Eq. 29, Eq. 30 and Eq. 31 represent an alternative algorithm for ILC on nonlinear systems when using a nonlinear ILC compensator. The essential difference is that the algorithm updates (a version of) the plant output instead of the plant input. The algorithm is shown in Fig. 2. Note that in the next section we will modify this algorithm and in the process ostensibly improve it, thus rendering the version presented here essentially a stepping stone or preliminary version. However, the research presented here focuses on implementing and evaluating the algorithm as presented in this section, with the evaluation of the modified version that is presented in the next section being the subject of future research.

4.2. Relationship with Conventional Algorithm

The system formulation in the iteration domain of the system input for the alternative algorithm is given as (combining Eq. 29, Eq. 30 and Eq. 31):

$$\begin{aligned} u^{(i+1)} &= L(\tilde{y}^{(i+1)}) \\ &= L(Q(\tilde{y}^{(i)} + y_d - y^{(i)})) \end{aligned} \quad (32)$$

$$= L(Q(L^{-1}(u^{(i)}) + y_d - T(u^{(i)}))) . \quad (33)$$

Clearly, when L is linear then Eq. 33 reduces to Eq. 6. In other words, if L is linear then the alternative algorithm is equivalent to the conventional algorithm, even if the system T is nonlinear. The alternative algorithm is therefore only an option when we employ a nonlinear inverse, and thus for ILC on nonlinear plants only. Note further that by a simple modification the input system formulation for the conventional algorithm (Eq. 6) may be restated as (using Eq. 30)

$$u^{(i+1)} = Q(u^{(i)} + L(y_d) - L(y^{(i)})) \quad (34)$$

$$= Q(L(\tilde{y}^{(i)}) + L(y_d) - L(y^{(i)})) \quad (35)$$

Comparison of Eq. 32 for $Q = 1$, i.e.

$$u^{(i+1)} = L(\tilde{y}^{(i)} + y_d - y^{(i)}) ,$$

and Eq. 35 for $Q = 1$, i.e.

$$u^{(i+1)} = L(\tilde{y}^{(i)}) + L(y_d) - L(y^{(i)}) ,$$

shows that the essential difference between the two ILC algorithms is that in the alternative algorithm L operates on all three entities together, instead of separately as in the conventional algorithm. This comes at no apparent cost as the alternative algorithm is still capable of achieving convergence in one iteration when $L = T^{-1}$ (as will be shortly shown).

4.3. Convergence

By inserting Eq. 29 and Eq. 30 into Eq. 31 the system formulation in the iteration domain for \tilde{y} is derived as

$$\tilde{y}^{(i+1)} = Q(\tilde{y}^{(i)} + y_d - T(L(\tilde{y}^{(i)}))) . \quad (36)$$

To analyze convergence we follow exactly the same approach as for the conventional algorithm, starting with the boundedness of the ILC algorithm.

Theorem 4.1: If T is BIBO stable in the 2-norm and $\alpha\bar{\beta}_0 < 1$ with $\alpha = \|Q\|_\infty$ and

$$\bar{\beta}_0 = \sup_{\tilde{y} \neq 0} \frac{\|\tilde{y} - T(L(\tilde{y}))\|_2}{\|\tilde{y}\|_2} , \quad (37)$$

then Eq. 31 is BIBO stable in the 2-norm. The proof is similar to that of Theorem 3.1.

◇

Next we focus on the existence of convergence points for the fixed point iteration of Eq. 36, i.e. the of limit signals. Noting Eq. 9 the limit signals for Eq. 31 and Eq. 36 are respectively

$$\tilde{y}^{(\infty)} = Q(\tilde{y}^{(\infty)} + y_d - y^{(\infty)}) \quad (38)$$

$$= Q(\tilde{y}^{(\infty)} + y_d - T(L(\tilde{y}^{(\infty)}))) \quad (39)$$

If the limit signals exist, then for $Q = 1$ clearly from Eq. 39 $y_d = T(L(\tilde{y}^{(\infty)}))$, implying $\tilde{y}^{(\infty)} = L^{-1}(T^{-1}(y_d)) = L^{-1}(u_d) = \tilde{y}_d$ (cf. Eq. 28), in turn $u^{(\infty)} = L(\tilde{y}^{(\infty)}) = L(\tilde{y}_d) = u_d$, and from Eq. 29 $y^{(\infty)} = T(u^{(\infty)}) = T(u_d) = y_d$. If $Q \neq 1$, then from Eq. 38

$$\tilde{y}^{(\infty)} = Q(1 - Q)^{-1}(y_d - y^{(\infty)}) , \quad (40)$$

$u^{(\infty)} = L(\tilde{y}^{(\infty)})$, and $y^{(\infty)} = T(u^{(\infty)})$. We have the following theorem:

Theorem 4.2: If the conditions of Theorem 3.4 hold and L is BIBO stable, then for $Q \neq 1$ Eq. 40 is BIBO stable in the 2-norm and $\tilde{y}^{(\infty)}$, $u^{(\infty)}$ and $y^{(\infty)}$ are well defined. If $Q = 1$ then $\tilde{y}^{(\infty)}$, $u^{(\infty)}$ and $y^{(\infty)}$ are well defined, with $\tilde{y}^{(\infty)} = \tilde{y}_d$, $u^{(\infty)} = u_d$ and with $y^{(\infty)} = y_d$. The proof for the case $Q = 1$ follows from the preceding discussion. The proof for case $Q \neq 1$ is similar to Theorem 3.2. ◇

The iteration domain formulation for the convergence error of \tilde{y} w.r.t. the limit signal $\tilde{y}^{(\infty)}$ is then (using Eq. 36 and Eq. 39)

$$\tilde{\Delta}_0^{(i+1)} = \tilde{y}^{(\infty)} - \tilde{y}^{(i+1)} \quad (41)$$

$$= Q(\tilde{y}^{(\infty)} + y_d - T(L(\tilde{y}^{(\infty)}))) - Q(\tilde{y}^{(i)} + y_d - T(L(\tilde{y}^{(i)})))$$

$$= Q(\tilde{y}^{(\infty)} - \tilde{y}^{(i)} - (T(L(\tilde{y}^{(\infty)})) - T(L(\tilde{y}^{(i)}))))$$

$$= Q\tilde{\Delta}_0^{(i)} - Q(T(L(\tilde{y}^{(\infty)})) - T(L(\tilde{y}^{(\infty)} - \tilde{\Delta}_0^{(i)}))) \quad (42)$$

$$= Q\tilde{\Delta}_0^{(i)} - Q\tilde{\Phi}_0(\tilde{y}^{(\infty)}, \tilde{\Delta}_0^{(i)}) , \quad (43)$$

with

$$\tilde{\Phi}_0(\tilde{y}^{(\infty)}, \tilde{\Delta}_0^{(i)}) := T(L(\tilde{y}^{(\infty)})) - T(L(\tilde{y}^{(\infty)} - \tilde{\Delta}_0^{(i)})) . \quad (44)$$

We have the following sufficient condition for convergence:

Theorem 4.3: If the ILC update formula Eq. 31 is BIBO stable, the limit signal of Eq. 39 is well defined, and $\alpha\tilde{\gamma}_0 < 1$ with $\alpha = \|Q\|_\infty$ and

$$\tilde{\gamma}_0 = \sup_{\tilde{\Delta}_0 \neq 0} \frac{\|\tilde{\Delta}_0 - \tilde{\Phi}_0(\tilde{y}^{(\infty)}, \tilde{\Delta}_0)\|_2}{\|\tilde{\Delta}_0\|_2}, \quad (45)$$

then $\|\tilde{\Delta}_0^{(i+1)}\| < \alpha\tilde{\gamma}_0\|\tilde{\Delta}_0^{(i)}\|$, that is, the convergence error of $\tilde{y}^{(i)}$ w.r.t. $\tilde{y}^{(\infty)}$ is monotonically decreasing and $\tilde{y}^{(i)}$ converges to $\tilde{y}^{(\infty)}$. The proof is similar to Theorem 3.3.

◇

Substitution of the GIB compensator (Eq. 18) and Eq. 44 into Eq. 45, gives

$$\tilde{\gamma}_0 = \sup_{\tilde{\Delta}_0 \neq 0} \frac{\|\tilde{\Delta}_0 - (T(L(\tilde{y}^{(\infty)})) - T(L(\tilde{y}^{(\infty)} - \tilde{\Delta}_0)))\|_2}{\|\tilde{\Delta}_0\|_2} \quad (46)$$

$$= \sup_{\tilde{\Delta}_0 \neq 0} \frac{\|\tilde{\Delta}_0 - (T(cC\tilde{L}(\tilde{y}^{(\infty)})) - T(cC\tilde{L}(\tilde{y}^{(\infty)} - \tilde{\Delta}_0)))\|_2}{\|\tilde{\Delta}_0\|_2} \quad (47)$$

When the convergence condition of Theorem 4.3, namely $\alpha\tilde{\gamma}_0 < 1$ is violated, the role of a small c in recovering convergence (at the expense of a slower rate of convergence) is clear, however its effect is not as obvious as in the case of the conventional algorithm.

For the ideal GIB compensator with $\tilde{L} = T^{-1}$ and $C = 1$, i.e. $L = cT^{-1}$, Eq. 42 becomes

$$\tilde{\Delta}_0^{(i+1)} = Q\tilde{\Delta}_0^{(i)} - Q(T(cT^{-1}(\tilde{y}^{(\infty)})) - T(cT^{-1}(\tilde{y}^{(\infty)} - \tilde{\Delta}_0^{(i)}))). \quad (48)$$

Clearly we do not have the same kind of simplification resulting from use of the ideal GIB compensator as we have for the conventional ILC algorithm (cf. Eq. 21). However, if $c = 1$ we again find $\tilde{\Delta}_0^{(i+1)} = 0$, i.e. convergence is still achieved in one iteration, with the limit signals as given in Theorem 4.2. While c will be effective in setting the rate of general convergence, the rate of convergence can not be as precisely controlled as for the conventional case due to the nonlinearity of T in Eq. 48.

4.4. Connection with Fixed Point Iteration Methods

The system formulation in the iteration domain for the GIB compensator (Eq. 18) with $c = 1$, and $Q = 1$ becomes (from Eq. 36)

$$\tilde{y}^{(i+1)} = \tilde{y}^{(i)} + y_d - T(cC\tilde{L}(\tilde{y}^{(i)})), \quad (49)$$

$$= \tilde{y}^{(i)} + y_d - T(C\tilde{L}(\tilde{y}^{(i)})) \quad (50)$$

$$:= \bar{\mathcal{T}}_{11}^0(\tilde{y}^{(i)}, y_d). \quad (51)$$

If we now formulate a standard Mann iteration scheme with the operator $\bar{T}_{11}^0(\tilde{y}^{(i)}, y_d)$, we get

$$\tilde{y}^{(i+1)} = (1 - \alpha_i)\tilde{y}^{(i)} + \alpha_i\bar{T}_{11}^0(\tilde{y}^{(i)}, y_d) \quad (52)$$

$$= (1 - \alpha_i)\tilde{y}^{(i)} + \alpha_i(\tilde{y}^{(i)} + y_d - T(C\tilde{L}(\tilde{y}^{(i)}))) \quad (53)$$

$$= \tilde{y}^{(i)} + \alpha_i y_d - \alpha_i T(C\tilde{L}(\tilde{y}^{(i)})) \quad (54)$$

with $\alpha_i \in (0, 1]$. Comparing Eq. 54 with Eq. 49 shows that performing ILC with $Q = 1$, with the alternative algorithm, and with the GIB compensator with $c \in (0, 1)$ is *not* equivalent to Mann iteration with the operator $\bar{T}_{11}^0(\tilde{y}^{(i)}, y_d)$ for the special case of using a constant $\alpha_i = c$. This will be true for the preliminary algorithm for any ILC compensator featuring a constant scale factor.

5. Modified Alternative ILC Algorithm Using a Nonlinear Inverse

5.1. Algorithm

In this section we modify the alternative algorithm presented in Section 3 in order to obtain it in a way that again allows precise control over the rate of convergence, straightforward adjustment of the convergence condition (γ) to ensure convergence, and equivalence between use of the GIB compensator for $c \in (0, 1)$ and Mann iteration with the operator obtained when using $c = 1$ in the GIB compensator. We consider the same nonlinear system and associated assumptions as in Section 3, and again retain the Q filter in an optional capacity. Define \tilde{y}_d such that

$$u_d = \tilde{L}(\tilde{y}_d), \quad (55)$$

with \tilde{L} as defined in the GIB compensator (Eq. 18), i.e. as representing the inverse model *per se*. Furthermore,

$$y^{(i)} = T(u^{(i)}), \quad (56)$$

and

$$u^{(i)} = \tilde{L}(\tilde{y}^{(i)}). \quad (57)$$

We modify the update formula of the preliminary alternative algorithm (Eq. 31) as follows:

$$\tilde{y}^{(i+1)} = Q(\tilde{y}^{(i)} + cC(y_d - y^{(i)})), \quad (58)$$

with initial value $\tilde{y}^{(0)} = [0]$, and c and C as in the GIB compensator. When $Q = 1$ clearly, when $y^{(i)}$ converges to y_d in Eq. 58, $\tilde{y}^{(i)}$ and $u^{(i)}$ converges to \tilde{y}_d and u_d respectively. Eq. 56, Eq. 57 and Eq. 58 represent a modified form of the alternative algorithm for ILC on nonlinear systems when using a nonlinear ILC compensator.

5.2. Relationship with Conventional Algorithm

The system formulation in the iteration domain of the system input for the modified alternative algorithm is given as (combining Eq. 56, Eq. 57 and Eq. 58):

$$\begin{aligned} u^{(i+1)} &= \tilde{L}(\tilde{y}^{(i+1)}) \\ &= \tilde{L}(Q(\tilde{y}^{(i)} + cC(y_d - y^{(i)}))) \end{aligned} \quad (59)$$

$$= \tilde{L}(Q(\tilde{L}^{-1}(u^{(i)}) + cC(y_d - T(u^{(i)})))) . \quad (60)$$

When \tilde{L} is linear Eq. 60 reduces to Eq. 6 and the modified alternative algorithm becomes equivalent to the conventional algorithm (even for nonlinear T), and is therefore only an option when we employ a nonlinear inverse.

Note further that by a simple modification the input system formulation for the conventional algorithm (Eq. 6) may be restated as (using Eq. 57)

$$u^{(i+1)} = Q(u^{(i)} + L(y_d) - L(y^{(i)})) \quad (61)$$

$$= Q(\tilde{L}(\tilde{y}^{(i)}) + cC(\tilde{L}(y_d) - \tilde{L}(y^{(i)}))) \quad (62)$$

Comparison of Eq. 59 for $Q = 1$, i.e.

$$u^{(i+1)} = \tilde{L}(\tilde{y}^{(i)} + cC(y_d - y^{(i)})) ,$$

and Eq. 62 for $Q = 1$, i.e.

$$u^{(i+1)} = \tilde{L}(\tilde{y}^{(i)}) + cC(\tilde{L}(y_d) - \tilde{L}(y^{(i)}))$$

shows that the essential difference between the two ILC algorithms is not just that in the modified alternative algorithm \tilde{L} operates on all three entities together, instead of separately as in the conventional algorithm, but is also the location of the cC factor w.r.t. \tilde{L} (i.e. “inside” \tilde{L} vs. “outside” \tilde{L}). This comes at no apparent cost as the alternative algorithm is still capable of achieving convergence in one iteration when $L = T^{-1}$ (as will be shortly shown).

5.3. Convergence

By inserting Eq. 56 and Eq. 57 into Eq. 58 the system formulation in the iteration domain for \tilde{y} is derived as

$$\tilde{y}^{(i+1)} = Q(\tilde{y}^{(i)} + cC(y_d - T(\tilde{L}(\tilde{y}^{(i)})))) . \quad (63)$$

To analyze convergence we again follow the same approach as for the conventional algorithm.

Theorem 5.1: If T is BIBO stable in the 2-norm and $\alpha\bar{\beta} < 1$ with $\alpha = \|Q\|_\infty$ and

$$\bar{\beta} = \sup_{\tilde{y} \neq 0} \frac{\|\tilde{y} - cCT(\tilde{L}(\tilde{y}))\|_2}{\|\tilde{y}\|_2} , \quad (64)$$

then Eq. 58 is BIBO stable in the 2-norm. The proof is similar to that of Theorem 3.1.

◇

Noting Eq. 9 the limit signals for Eq. 58 and Eq. 63 are defined by respectively

$$\tilde{y}^{(\infty)} = Q(\tilde{y}^{(\infty)} + cC(y_d - y^{(\infty)})) \quad (65)$$

$$= Q(\tilde{y}^{(\infty)} + cC(y_d - T(\tilde{L}(\tilde{y}^{(\infty)})))) . \quad (66)$$

If the limit signals exist, then for $Q = 1$ clearly from Eq. 66 $y_d = T(\tilde{L}(\tilde{y}^{(\infty)}))$, implying $\tilde{y}^{(\infty)} = \tilde{L}^{-1}(T^{-1}(y_d)) = \tilde{L}^{-1}(u_d) = \tilde{y}_d$ (cf. Eq. 55), in turn $u^{(\infty)} = \tilde{L}(\tilde{y}^{(\infty)}) = \tilde{L}(\tilde{y}_d) = u_d$, and $y^{(\infty)} = T(u^{(\infty)}) = T(u_d) = y_d$. If $Q \neq 1$, then from Eq. 65

$$\tilde{y}^{(\infty)} = cCQ(1 - Q)^{-1}(y_d - y^{(\infty)}) , \quad (67)$$

$u^{(\infty)} = L(\tilde{y}^{(\infty)})$, and $y^{(\infty)} = T(u^{(\infty)})$. We have the following theorem:

Theorem 5.2: If the conditions of Theorem 3.7 hold and L is BIBO stable, then for $Q \neq 1$ Eq. 67 is BIBO stable in the 2-norm and $\tilde{y}^{(\infty)}$, $u^{(\infty)}$ and $y^{(\infty)}$ well defined. If $Q = 1$ then $\tilde{y}^{(\infty)}$, $u^{(\infty)}$ and $y^{(\infty)}$ are well defined, with $\tilde{y}^{(\infty)} = \tilde{y}_d$, $u^{(\infty)} = u_d$ and with $y^{(\infty)} = y_d$. The proof for the case $Q = 1$ follows from the preceding discussion. The proof for case $Q \neq 1$ is similar to Theorem 3.2. \diamond

The iteration domain formulation for the convergence error of the input signal w.r.t. the limit signal is then (using Eq. 63 and Eq. 66)

$$\tilde{\Delta}^{(i+1)} = \tilde{y}^{(\infty)} - \tilde{y}^{(i+1)} \quad (68)$$

$$= Q(\tilde{y}^{(\infty)} + cC(y_d - T(\tilde{L}(\tilde{y}^{(\infty)})))) - Q(\tilde{y}^{(i)} + cC(y_d - T(\tilde{L}(\tilde{y}^{(i)}))))$$

$$= Q(\tilde{y}^{(\infty)} - \tilde{y}^{(i)} - cC(T(\tilde{L}(\tilde{y}^{(\infty)})) - T(\tilde{L}(\tilde{y}^{(i)}))))$$

$$= Q\tilde{\Delta}^{(i)} - QcC(T(\tilde{L}(\tilde{y}^{(\infty)})) - T(\tilde{L}(\tilde{y}^{(\infty)} - \tilde{\Delta}^{(i)}))) \quad (69)$$

$$= Q\tilde{\Delta}^{(i)} - Q\tilde{\Phi}(\tilde{y}^{(\infty)}, \tilde{\Delta}^{(i)}) , \quad (70)$$

with

$$\tilde{\Phi}(\tilde{y}^{(\infty)}, \tilde{\Delta}^{(i)}) := cC(T(\tilde{L}(\tilde{y}^{(\infty)})) - T(\tilde{L}(\tilde{y}^{(\infty)} - \tilde{\Delta}^{(i)}))) . \quad (71)$$

We have the following sufficient condition for convergence:

Theorem 5.3: If the ILC update formula Eq. 58 is BIBO stable, the limit signal of Eq. 66 is well defined, and $\alpha\tilde{\gamma} < 1$ with $\alpha = \|Q\|_\infty$ and

$$\tilde{\gamma} = \sup_{\tilde{\Delta} \neq 0} \frac{\|\tilde{\Delta} - \tilde{\Phi}(\tilde{y}^{(\infty)}, \tilde{\Delta})\|_2}{\|\tilde{\Delta}\|_2} , \quad (72)$$

then $\|\tilde{\Delta}^{(i+1)}\| < \alpha\tilde{\gamma}\|\tilde{\Delta}^{(i)}\|$, that is, the convergence error of $\tilde{y}^{(i)}$ w.r.t. $\tilde{y}^{(\infty)}$ is monotonically decreasing and $\tilde{y}^{(i)}$ converges to $\tilde{y}^{(\infty)}$. The proof is similar to Theorem 3.3. \diamond

Substituting Eq. 71 into Eq. 72, giving

$$\tilde{\gamma} = \sup_{\tilde{\Delta} \neq 0} \frac{\|\tilde{\Delta} - cC(T(\tilde{L}(\tilde{y}^{(\infty)})) - T(\tilde{L}(\tilde{y}^{(\infty)} - \tilde{\Delta})))\|_2}{\|\tilde{\Delta}\|_2} . \quad (73)$$

When the convergence condition of Theorem 5.3, namely $\alpha\gamma < 1$ is violated, the role of a small c in recovering convergence (at the expense of a slower rate of convergence) is again clear.

For the ideal inverse compensator $\tilde{L} = T^{-1}$ and $C = 1$ Eq. 69 becomes

$$\begin{aligned} \tilde{\Delta}^{(i+1)} &= Q\tilde{\Delta}^{(i)} - Qc(T(T^{-1}(\tilde{y}^{(\infty)})) - T(T^{-1}(\tilde{y}^{(\infty)} - \tilde{\Delta}^{(i)}))) \\ &= Q(1 - c)\tilde{\Delta}^{(i)}. \end{aligned} \tag{74}$$

Clearly we again have the same kind of simplification resulting from use of the ideal GIB compensator as we have for the conventional ILC algorithm (cf. Eq. 21). Furthermore, if $c = 1$ we again find $\tilde{\Delta}^{(i+1)} = [0]$, i.e. convergence is still achieved in one iteration, with the limit signals as given in Theorem 5.2. In addition the rate of convergence can be precisely controlled as in the conventional case.

5.4. Connection with Fixed Point Iteration Methods

The system formulation in the iteration domain for $Q = 1$ becomes (from Eq. 63)

$$\tilde{y}^{(i+1)} = \tilde{y}^{(i)} + cC(y_d - T(\tilde{L}(\tilde{y}^{(i)}))), \tag{75}$$

and for $c = 1$ the system formulation then becomes

$$\tilde{y}^{(i+1)} = \tilde{y}^{(i)} + C(y_d - T(\tilde{L}(\tilde{y}^{(i)}))) \tag{76}$$

$$:= \bar{\mathcal{T}}_{11}(\tilde{y}^{(i)}, y_d). \tag{77}$$

If we now formulate a standard Mann iteration scheme with the operator $\bar{\mathcal{T}}_{11}(\tilde{y}^{(i)}, y_d)$, we get

$$\begin{aligned} \tilde{y}^{(i+1)} &= (1 - \alpha_i)\tilde{y}^{(i)} + \alpha_i\bar{\mathcal{T}}_{11}(\tilde{y}^{(i)}, y_d) \\ &= (1 - \alpha_i)\tilde{y}^{(i)} + \alpha_i(\tilde{y}^{(i)} + C(y_d - T(\tilde{L}(\tilde{y}^{(i)})))) \\ &= \tilde{y}^{(i)} + \alpha_iC(y_d - T(\tilde{L}(\tilde{y}^{(i)}))) \end{aligned} \tag{78}$$

with $\alpha_i \in (0, 1]$. Comparing Eq. 78 with Eq. 75 shows that performing ILC with the modified alternative algorithm with $Q = 1$ and $c \in (0, 1)$ is equivalent to Mann iteration with the operator $\bar{\mathcal{T}}_{11}(\tilde{y}^{(i)}, y_d)$ (for the special case of a constant α).

6. Example 1: ILC for a Deterministic Desired Signal

This example demonstrates the ability of the alternative and conventional ILC algorithms to converge for a short-duration deterministic signal. The test system is represented by

$$\begin{aligned} y(k) &= \theta_1u(k - 4) + \theta_2u(k - 5) + \theta_3u(k - 6) + \theta_4y(k - 4) \\ &\quad \theta_5u(k - 5)y(k - 4) + \theta_6u(k - 5)u(k - 6)y(k - 2) \\ &\quad + \theta_7u(k - 5)^2u(k - 6)y(k - 1), \end{aligned} \tag{79}$$

with

$$(\theta_1, \dots, \theta_7) = (0.150, 0.50, 0.50, 1/6, -2.0, 6.0, 11.0) .$$

The desired response in question is obtained as the response of Eq. 79 to the following relatively short deterministic signal, $u_d(k)$:

$$\bar{u}_d(k) = \begin{cases} 0, & 1 \leq k \leq 25 \\ c_u(\sin(2\pi(k - 31)/20) + 1), & 25 < k \leq 46 \\ 0, & 46 < k \leq 146 \end{cases} \quad (80)$$

$$\tilde{u}_d = \bar{F}_{0.16}\bar{u}_d \quad (81)$$

$$u_d = C_T\tilde{u}_d \quad (82)$$

$$F_{0.16}(z) = \frac{0.02287z^4 + 0.09148z^3 + 0.13722z^2 + 0.09148z + 0.02287}{1.00z^4 - 1.412z^3 + 1.123z^2 - 0.40807z + 0.06321} \quad (83)$$

$$C_T = \begin{cases} 0, & 1 \leq k \leq 12 \\ 0.5 \sin(2\pi(k - 13)/24 - \pi/2) + 0.5, & 12 < k \leq 24 \\ 1, & 24 < k \leq 122 \\ 0.5 \sin(2\pi(k - 123)/24 + \pi/2) + 0.5, & 122 < k \leq 134 \\ 0, & 134 < k \leq 146 \end{cases} \quad (84)$$

with $c_u = 0.165$, the second equation in operator format, and \bar{F} a non-causal linear operator representing the zero phase version of the low pass filter $F(z)$ with cut frequency 40 Hz (0.16 times the sample frequency). C_T as given by Eq. 84 is essentially a sinusoidal taper function. The desired input signal and desired response signal is shown in Fig. 4. For purposes of obtaining the inverse-model based ILC compensator we obtained the following approximate (nonlinear) NARX system model by system identification:

$$\begin{aligned} y(k) = & -0.076129u(t - 2) + 0.21444u(t - 3) - 0.0040361u(t - 10) \\ & + 2.6585y(t - 1) - 3.6336y(t - 2) + 3.2178y(t - 3) \\ & - 1.8455y(t - 4) + 0.56977y(t - 5) - 0.039669y(t - 7) \\ & + 0.12012u(k - 3)u(k - 4) - 0.12388u(k - 3)u(k - 10) \\ & - 0.12629u(k - 3)y(k - 2) . \end{aligned} \quad (85)$$

Due to the nonlinearity of the model, stable inversion of the model is iterative. A gain of 0.1 was used in stable inversion. For ILC a zero-phase low pass ILC filter Q with cut frequency of 40 Hz was used and a ILC gain of $c = 0.3$ was used. The results for both the conventional and alternative ILC algorithms are presented in Fig. 5 and Fig. 6. The error between $u_d(k)$ and the input calculated during iteration m of stable inversion, namely $u^{(m)}(k)$, is defined as:

$$\begin{aligned} \text{err}_1(u^{(m)}) & := 100 \frac{\sum_{k=1}^N |u^{(m)}(k) - u_d(k)|}{\sum_{k=1}^N |u_d(k)|} \\ & = 100 \frac{\|u^{(m)}(k) - u_d(k)\|_1}{\|u_d(k)\|_1} . \end{aligned} \quad (86)$$

Similarly

$$\text{err}_1(y^{(m)}) := 100 \|y^{(m)}(k) - y_d(k)\|_1 / \|y_d(k)\|_1 . \quad (87)$$

Clearly both the conventional and alternative ILC algorithms converge with the alternative algorithm converging more rapidly and to lower convergence errors values. The minimum convergence error values in these figures correspond to virtually imperceptible differences between the desired and achieved input and response signals. Note that sinusoidal tapering (of duration 0.03 sec) was applied during stable inversion during the alternative ILC algorithm to suppress build-up of spurious high frequency oscillations at the start of the signal.

7. Example 2: ILC for a Random Desired Signal

In this example ILC is performed on Eq. 79 using the low-level random input and corresponding output signal of Example 2 in [13], shown in Fig. 7, to demonstrate the ability of the alternative ILC algorithm to converge when the conventional algorithm diverges in this particular case. For the ILC compensator the stable inverse of Eq. 85 was again used, this time with a stable inversion gain of 0.2. An ILC gain of $c = 0.4$ was used and a zero-phase low pass ILC filter Q was used with a cut frequency 50 Hz.

The results for both the conventional and alternative ILC algorithms are presented in Fig. 8 and Fig. 9, which shows that the conventional ILC algorithm diverges and alternative ILC algorithm converges. The approach here (as in Example 1) of using the Q filter and not the C filter to achieve the low pass frequency cut off for ILC follows from the fact that that is the intention of the Q filter in normal practise, whereas the C filter is rather intended to be a *shaping* filter used to increase the bandwidth of convergence.

8. Example 3: ILC Using an Approximate Linear Inverse Model

In this example ILC is performed on Eq. 79 using the higher-level random input and corresponding output signal of Example 3 in [13], shown in Fig. 10. The purpose is to demonstrate the success of ILC using a linear inverse model-based ILC compensator. This doesn't imply that ILC converges, in fact it diverges in this case, and from a purely ILC point of view that will be a drawback, with the approach to remedy this to tailor the Q and C filters and the c gain to obtain the largest bandwidth possible that still gives convergence of ILC. However, in response reconstruction for structural integrity testing we do not have a purely ILC point of view that requires convergence at all costs, but rather we *use* ILC for a limited number of iterations to obtain the most accurate results possible over the largest bandwidth possible (or over a fixed, given bandwidth), with ILC terminated as soon as the most accurate results have been obtained. Thus, in response reconstruction it is tolerable if ILC diverges, as long as the results obtained before divergence is sufficiently accurate for the purposes of the test.

A zero-phase, low pass ILC filter Q is employed, with the cut frequency of Q respectively 50, 70, 90 Hz, including the option of using no filter. While there is very little signal strength in the desired response above 50 Hz, it is very much the higher frequencies (50 Hz and above) that are responsible for divergence of ILC in this case. It is therefore sensible to investigate the use of low pass ILC Q filters with cut frequencies of 50 Hz and higher to investigate the reduction in the severity of divergence while not severely reducing the signal frequency range.

In order to systematically evaluate ILC for the various values of Q a range of ILC gains will be used for every value of Q , including both iteration independent and iteration dependent gains, from which the best results may subsequently be selected. The formula for the iteration dependent gain $c^{(i)}$ is given as

$$c^{(i)} = 2 \frac{c_0 - c_{\text{lim}}}{i + 1} + c_{\text{lim}}, \quad (88)$$

noting that $c^{(0)} = c_0$.

For purposes of the inverse-model based ILC compensator a linear inverse model was obtained by performing system identification, and inverting the resulting model via stable inversion. Stable inversion of a linear model is accomplished in a single pass, and is exact (in the sense that the calculated input signal, when passed back through the model, recovers the given output signal used in the inversion). The following (linear) ARX model of the system was obtained (prior to inversion):

$$\begin{aligned} y(k) = & -0.0020592u(k-4) + 0.53042u(k-5) - 0.33295u(k-6) + 2.9341y(k-1) \\ & - 4.5461y(k-2) + 4.6988y(k-3) - 3.3494y(k-4) + 1.542y(k-5) \\ & - 0.34179y(k-6). \end{aligned} \quad (89)$$

The best results of ILC on Eq. 79 using the stable inverse of this model is presented in Table 1. Note that when the ILC compensator L is linear, the conventional and alternative ILC algorithms are equivalent, for which reason we do not distinguish between the two approaches here. The best input signal that was obtained had an error of 18.0%, and the corresponding output, that is obtained by passing the input back through Eq. 79 (i.e. the system), had an error of 9.9%. These results do not represent an improvement on the results of stable inversion Eq. 79 for the same desired input and output signal in Example 3 of [13], for which we get lower input and output errors of 13.3% and 8.1% respectively. It is found that in all cases except when using no Q filter the best ILC results were obtained using iteration-dependent ILC gains.

Table 1. Example 3: Best results of ILC. M is the iteration resulting in $\min_m \text{err}_1(u^{(m)})$, i.e. $M = \text{argmin}_m \text{err}_1(u^{(m)})$.

Q-filter cut freq. [Hz]	$\min_m \text{err}_1(u^{(m)})$ [%]	$\text{err}_1(y^{(M)})$ [%]	M = Iter. no.	Comment
50	18.0	9.9	432	$c_0 = 0.1$; $c_{\text{lim}} = 0.01$ (it. var.)
70	21.4	11.4	86	$c_0 = 0.3$; $c_{\text{lim}} = 0.025$ (it. var.)
90	41.1	29.4	78	$c_0 = 0.1$; $c_{\text{lim}} = 0.01$ (it. var.)
None	52.3	75.0	1	$c = 1$ for u ; $c = 0.4$ for y

9. Example 4: ILC Using an Approximate Nonlinear Inverse Model

In this example, as in Example 3, ILC is performed on Eq. 79 using the higher level random input and corresponding output signal of Example 3, shown in Fig. 10. However, for the inverse model-based ILC compensator we use here a nonlinear inverse model for purposes of comparison of the results with those of Example 3. For the nonlinear inverse

model the following approximate NARX model was obtained by system identification:

$$\begin{aligned}
y(k) = & 0.23043u(t-4) + 0.10461u(t-6) - 0.037279u(t-10) \\
& + 1.7955y(t-1) - 2.0957y(t-2) + 1.7392y(t-3) \\
& - 0.88389y(t-4) + 0.24564y(t-5) \\
& - 0.29621u(k-5)y(k-5) - 1.2585u(k-5)^3 \\
& + 7.2289u(k-5)^2y(k-1) - 9.2816u(k-5)u(k-6)y(k-2) \\
& + 4.4206u(k-5)u(k-7)y(k-3) - 0.30189u(k-10)y(k-2)y(k-9) \\
& + 0.041543y(k-1)^2y(k-10) .
\end{aligned} \tag{90}$$

The model is not particularly optimized, but is representative of the type of nonlinear models obtained in response reconstruction using automated methods for searching for best models over a range of model structures.

Since the model is nonlinear, stable inversion of the model is iterative. Since the iteration of stable inversion in this case is not convergent, regardless of the choice of gain in stable inversion, a rough optimization was done to determine the choice of gain in stable inversion that gives stable inversion results comparable to the best possible results. This was deemed sufficient as the desired input is usually not available in practise to determine inversion accuracy, systematic optimization can be an intensive exercise and often does not result in significant gains over the results obtained with rough optimization, and ILC does not critically depend on the accuracy of the inverse but is capable of iterative correction in the calculation of the input, i.e. is somewhat robust against inaccuracies in the accuracy of the inverse model (within limits of course). The number of iterations in stable inversion was limited to about 1000 iterations (and more where more was needed before divergence occurred).

ILC is evaluated in this case for various combinations of ILC low pass filters Q and ILC gains c , and for both the conventional and alternative ILC algorithms. The best results obtained using the conventional ILC algorithm is shown in Table 2, and for the alternative ILC algorithm in Table 3. Negative ILC gains c in Table 2 and Table 3 implies that an iteration dependent ILC gain was used with $c_0 = |c|$ and $c_{\text{lim}} = 0.005$ (cf. Eq. 88). Only time-independent ILC gains were used.

Results are presented for both iteration-independent and iteration dependent stable inversion gains, indicated as stable inversion cases A and B respectively, and for time-independent and time dependent stable inversion gains, indicated as stable inversion cases 1 and 2 respectively. Refer to Table 4. The formula for iteration dependent stable inversion gains used here is the same as for iteration dependent ILC gains, namely Eq. 88, with $c_0 = c_{\text{SI}}$ and $c_{\text{lim}} = c_{\text{lim,SI}}$. The time-dependent stable inversion gain for stable inversion iteration j and input channel i , designated $C_{v,i}^{(j)}(k)$, was calculated here by the following formula

$$\begin{aligned}
\bar{C}_{v,i}^{(j)}(k) &= 1/(800|u^{(j-1)}(k) - u^{(j)}(k)| + 1) \\
\tilde{C}_{v,i}^{(j)} &= \bar{F}_{0.083}\bar{C}_{v,i}^{(j)} - 1 \\
C_{v,i}^{(j)}(k) &= \frac{\tilde{C}_{v,i}^{(j)}(k)}{\max_k |\tilde{C}_{v,i}^{(j)}(k)|} + 1
\end{aligned} \tag{91}$$

with the second formula in operator format, and $\bar{F}_{0.083}$ a zero-phase low pass filter (with cut frequency 0.083 of the sample frequency). The $C_{v,i}^{(j)}(k)$ values were suitably delayed for application to the columns of $\eta^{(j)}(k)$. $C_{v,i}^{(j)}(k)$ is multiplied with the time-independent stable inversion gain for iteration j , which for the iteration dependent case is designated $c_{\text{SI}}^{(j)}$ (and is governed by the c_{SI} and $c_{\text{lim,SI}}$ values - cf. Table 4).

It is found that both the conventional and alternative ILC algorithms are consistently divergent. The best results were achieved with the alternative ILC algorithm using a 50 Hz low pass filter in Q for stable inversion case B2 (cf. Table 3), and was much more accurate than the results achieved with the conventional ILC algorithm.

None of the results in Table 2 and Table 3 represent an improvement over the results achieved in Example 3 with the linear inverse model for the same mathematical system and desired input and output signals. However, when we repeat the conventional and alternative ILC algorithm tests for the case of a 50 Hz low pass filter in Q and a 50 Hz low pass filter in stable inversion (see the results presented in Table 5), we do observe an improvement over the results in Example 3. The best results are consistently obtained with the time-dependent gain approach in stable inversion, with the conventional ILC algorithm giving slightly better results than the alternative algorithm.

Table 2. Example 4: Best ILC results with the conventional ILC algorithm. A negative ILC gain c implies an iteration-dependent ILC gain with $c_0 = |c|$ and $c_{\text{lim}} = 0.005$ (cf. Eq. 88). A positive c implies an iteration-independent ILC gain. “SI” refers to stable inversion. “It.” for u and y are the iteration numbers m corresponding to $\min_m \text{err}_1(u^{(m)})$ and $\min_m \text{err}_1(y^{(m)})$ respectively.

Q-filter cut freq ..[Hz]	Input - u			Output - y			SI Case
	c	It.	$\min_m \text{err}_1(u^{(m)})$ [%]	c	It.	$\min_m \text{err}_1(y^{(m)})$ [%]	
50	-0.3	48	52.06	-0.3	48	58.31	A1
70	-0.1	156	62.07	-0.1	156	68.77	A1
90	1.0	1	73.68	0.1	12	80.86	A1
None	-0.1	101	72.87	-0.1	101	73.88	A1
50	-0.1	199	50.22	-0.1	199	58.18	A2
70	1.0	1	65.01	-0.1	69	78.64	A2
90	0.05	25	65.08	0.05	25	72.70	A2
None	1.0	1	71.97	-0.1	41	78.68	A2
50	-0.1	411	48.81	-0.1	411	50.31	B1
70	-0.1	129	68.68	-0.1	129	75.03	B1
90	1.0	1	74.31	0.1	11	82.42	B1
None	1.0	1	75.86	-0.1	60	80.42	B1
50	1.0	1	46.26	-0.1	100	57.51	B2
70	1.0	1	47.22	-0.1	70	59.88	B2
90	1.0	1	48.83	-0.1	56	63.79	B2
None	1.0	1	50.52	-0.7	1	67.45	B2

Table 3. Example 4: Best ILC results with the alternative ILC algorithm. A negative ILC gain c implies an iteration-dependent ILC gain with $c_0 = |c|$ and $c_{\text{lim}} = 0.005$ (cf. Eq. 88). A positive c implies an iteration-independent ILC gain. “SI” refers to stable inversion. “It.” for u and y are the iteration numbers m corresponding to $\min_m \text{err}_1(u^{(m)})$ and $\min_m \text{err}_1(y^{(m)})$ respectively.

Q-filter cut freq ..[Hz]	Input - u			Output - y			SI Case
	c	It.	$\min_m \text{err}_1(u^{(m)})$ [%]	c	It.	$\min_m \text{err}_1(y^{(m)})$ [%]	
50	-0.1	360	63.45	-0.3	118	69.44	A1
70	1.0	1	75.36	1.0	1	81.06	A1
90	-0.1	154	72.16	-0.1	154	78.28	A1
None	-0.7	1	74.86	-0.7	1	81.07	A1
50	-0.1	238	38.69	-0.1	238	48.87	A2
70	-0.1	238	59.72	-0.1	238	66.94	A2
90	-0.1	129	59.66	-0.1	129	68.85	A2
None	0.4	1	66.50	0.4	1	74.69	A2
50	-0.1	383	59.11	-0.1	383	66.13	B1
70	-0.1	246	73.70	-0.7	2	78.98	B1
90	-0.7	2	76.23	-0.1	27	81.07	B1
None	0.1	8	75.17	0.1	8	80.63	B1
50	-0.3	100	21.70	-0.3	100	15.94	B2
70	-0.1	171	34.05	-0.1	171	32.68	B2
90	0.4	1	71.75	0.4	1	78.83	B2
None	1.0	1	50.52	0.05	10	74.63	B2

Table 4. Example 4: Stable inversion case details. “SI” refers to stable inversion.

Case	Iteration dependent?	c_{SI}	$c_{\text{lim,SI}}$	Time dependent?
A1	No	0.01	–	No
A2	No	0.03	–	Yes
B1	Yes	0.10	0.005	No
B2	Yes	0.30	0.100	Yes

Table 5. Example 4: Best ILC results with the conventional and alternative ILC algorithms when employing a 50 Hz low pass filter in stable inversion. A negative ILC gain c implies an iteration-dependent ILC gain with $c_0 = |c|$ and $c_{lim} = 0.005$ (cf. Eq. 88). A positive c implies an iteration-independent ILC gain. "SI" refers to stable inversion. "It." for u and y are the iteration numbers m corresponding to $\min_m \text{err}_1(u^{(m)})$ and $\min_m \text{err}_1(y^{(m)})$ respectively.

Q-filter cut freq ... [Hz]	Input - u			Output - y			SI Case	ILC algorithm
	c	It.	$\min_m \text{err}_1(u^{(m)})$ [%]	c	It.	$\min_m \text{err}_1(y^{(m)})$ [%]		
50	1.0	1	15.82	1.0	1	16.78	A1	Conv.
50	1.0	1	13.47	-0.3	753	7.40	A2	Conv.
50	1.0	1	15.64	1.0	1	20.22	B1	Conv.
50	1.0	1	14.47	-0.3	240	9.95	B2	Conv.
50	1.0	1	14.88	-1.0	9	10.37	A1	Alt.
50	1.0	2	14.42	-1.0	9	8.46	A2	Alt.
50	1.0	1	14.78	0.4	4	17.29	B1	Alt.
50	1.0	2	15.50	-1.0	8	8.65	B2	Alt.

10. Some Recommendations for Future Research

- The performance of the modified alternative ILC algorithm needs to be compared to that of the alternative and conventional algorithms in a similar way to examples 1 to 4 and in physical application of ILC in response reconstruction.
- This research showed that the use of Mann and Ishikawa iteration in stable inversion has an analogy in ILC in that the conventional and alternative ILC algorithms developed here both have parallels in the Picard and Mann iteration schemes (when using the GIB compensator). It is furthermore shown that the application of Ishikawa iteration to ILC results in potentially novel ILC iteration schemes for both the conventional and alternative ILC algorithms. The resulting ILC schemes need to be further researched for connections with existing ILC schemes in literature, and need to be performance tested.
- ILC in this research was evaluated with either constant or iteration dependent gains. The possible contribution of time-dependent ILC gains still need to be evaluated in theoretical examples and practical implementation in response reconstruction, particularly for the algorithms that have been developed in this research.
- The models in the stable inversion in this research are all analytical (i.e. smooth). ILC need to be investigated for purposes of inverting non-smooth models. (Such model may be obtained by system identification but are difficult to invert.) This may be done by approximating such models with smooth models (e.g. NARX models) and then using the smooth approximate models in stable inversion in ILC that is done on the non-smooth model as if on a physical system. If successful this may improve the accuracy of ILC on non-smooth physical systems by virtue of the better accuracy of the ILC-based inversion of identified non-smooth models of such systems than of stable inversion of smooth approximate models of such systems. This implies the use of ILC for purposes of inversion in the ILC compensator of ILC on a system, i.e. a nested ILC.

11. Conclusions

Nonlinear, inverse model-based ILC compensators that are solved directly by stable inversion were presented. We also showed how an accurate (nonlinear) inverse model in the ILC compensator simplifies the ILC dynamics, resulting in monotone convergence of ILC and allowing better control of the rate of convergence.

An alternative ILC algorithm was developed, and was demonstrated to have the same desirable properties as the conventional algorithm with regard to convergence and control of the rate of convergence.

We also noted that the use of Mann and Ishikawa iteration in stable inversion has an analogy in ILC in that the conventional and alternative ILC algorithms developed here both have parallels in the Picard and Mann iteration schemes.

In Example 2 we presented an example where the alternative ILC algorithm converges while the conventional algorithm diverges for a random signal. In Example 3 we perform ILC using the inverse of a linear approximate model. In Example 4 we perform ILC using the inverse of a nonlinear approximate model, leading to moderate but significant improvements compared to the results obtained in Example 3. When not using a filter in stable inversion the alternative ILC algorithm was much more accurate than the conventional algorithm. The NARX models used in these examples, particularly Example 4, are typical of the type of models obtained in nonlinear system identification as used in response reconstruction.

12. Funding

The financial assistance of the National Research Foundation and Investmech, Pty. Ltd. is gratefully acknowledged.

References

- [1] Ahn, H-S, Chen, Y, Moore, K.L. Iterative learning control: brief survey and categorization. *IEEE Transactions On Systems, Man, And Cybernetics*; 2007:37(6):1099-1121.
- [2] Arimoto, S, Kawamura, S, Miyazaki, F. Bettering operation of robots by learning. *Journal of Robotic Systems*; 1984:1(2):123-140.
- [3] Craig, JJ. Adaptive control of manipulators through repeated trials. *IEEE American Control Conference*; 1984:1566-1573.
- [4] Markusson, O. Model and system inversion with applications in nonlinear system identification and control, Doctoral thesis, Royal Institute of Technology, Stockholm, Sweden; 2002.
- [5] Ghosh, J, Paden, B. Iterative learning control for nonlinear nonminimum phase plants. *ASME Journal of Dynamic Systems, Measurement and Control*; 2001:123:21-30.
- [6] Ghosh, J, Paden, B. A pseudoinverse-based iterative learning control. *IEEE Trans. on Automatic Control*; 2002:47(5):831-837.
- [7] Smolders, K, Volckaert, M, Swevers, J. Tracking control of nonlinear lumped mechanical continuous-time systems: a model-based iterative learning approach. *Mechanical Systems and Signal Processing*; 2008:22:1896-1916.
- [8] Burden, RL, Faires, JD. Numerical analysis. 5th ed. PWS Publishing Company, Boston; 1993.
- [9] Lin, T, Owens, DH, HÄTÖNEN, J. Newton method based iterative learning control for discrete non-linear systems. *Int. J. of Control*; 2006:79(10):1263-1276.
- [10] Devasia, S, Chen, D, Paden, B. Nonlinear inversion-based output tracking. *IEEE Trans. on Automatic Control*; 1996:AC-41:(7):930-942.
- [11] Zeng, G, Hunt, LR. Stable inversion for nonlinear discrete-time systems. *IEEE Trans. on Automatic Control*; 2000:45(6):1216-1220.
- [12] Mann, WR. Mean value in iteration. *Proc. of the Am. Math. Soc.*; 1953:4:506-510.
- [13] Eksteen, JJA, Heyns, PS. Improvements in Stable Inversion of NARX Models by Using Mann Iteration. Submitted to *Inverse Problems in Science and Engineering*; 2014.
- [14] Jang, HS, Longman, RW. A new learning control law with monotonic decay of the tracking error norm. *Proc. of the Thirty-Second Annual Allerton Conf. on Communication, Control and Computing, Coordinated Sciences Laboratory, University of Illinois at Urbana-Champaign, USA*; 1994:314-323.
- [15] Jang, HS, Longman, RW. An update on a monotonic learning control law and some fuzzy logic learning gain adjustment techniques. *Advances in the Astronautical Sciences*; 1996a:90:301-318.
- [16] Jang, HS, Longman, RW. Design of digital learning controllers using a partial isometry. *Advances in the Astronautical Sciences*; 1996b:93:137-152.
- [17] Elci, H, Longman, RW, Phan, M, Juang, J-N, Ugoletti, R. Automated learning control through model updating for precision motion control. *ASME J. of Adaptive Structures and Composite Materials: Analysis and Application*; 1994b:AD-45/MD:54:299-314.
- [18] Ghosh, J, Paden, B. Pseudo-inverse based iterative learning control for linear nonminimum phase plants with unmodeled dynamics. *ASME Journal of Dynamic Systems, Measurement and Control*; 2004:126:661-665.
- [19] Phan, M, Longman, RW. Indirect learning control with guaranteed stability. *Proc. of the 1989 Conf. on Information Sciences and Systems, John Hopkins University, Baltimore, MD*; 1989:125-131.
- [20] Lee-Glauser, G, Juang, J-N, Longman, RW. Comparison and combination of learning controllers: computational enhancement and experiments. *Journal of Guidance, Control and*

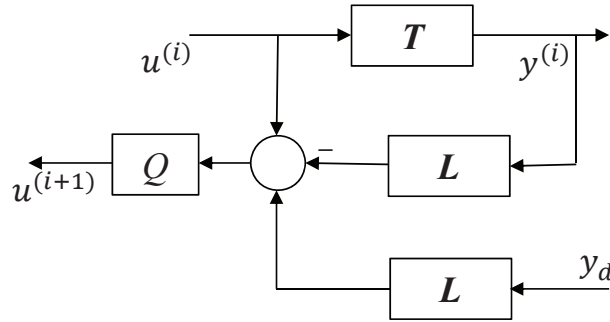


Figure 1. Conventional ILC algorithm on a nonlinear system T with general (possibly nonlinear) L and (optional) zero phase filter Q .

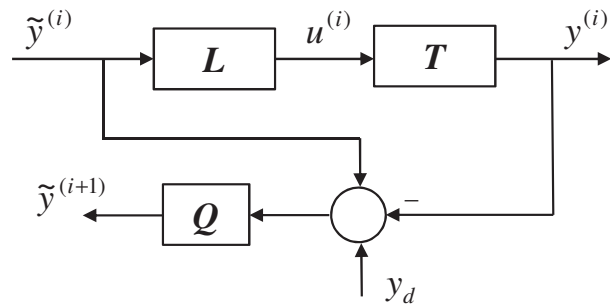


Figure 2. Alternative algorithm for ILC on a nonlinear system T using a nonlinear L .

Dynamics; 1996:19:1116-1123.

[21] Cryer, BW, Nawrocki, PE, Lund, RA. A road simulation system for heavy duty vehicles. SAE Transactions, SAE paper 760361; 1976:1322-1334.

[22] Gunnarsson, S, Norrlöf, M. On the use of learning control for improved performance in robot control systems. Proc. of the European control conf. on Decision and Control; 1997:586-591.

[23] Norrlöf, M., Gunnarsson, S. Time and frequency domain convergence properties in iterative learning control. Int. J. Contr.; 2002:75(14):1114-1126.

[24] Weal, P, Liefoghe, C, Dressler, K. Product durability engineering - improving the process. Sound and Vibration, 30th Anniversary Issue, January; 1997:68-79.

[25] Raath, AD. Service load simulation testing in the time domain. Environmental Engineering; 1993:6(3):8-16.

[26] Raath, AD. A new time domain parametric dynamic system identification approach to multiaxial service load simulation testing in components. Int. Journal of Fatigue; 1997:19(5):409-414.

[27] De Cuyper, J, Coppens, D, Liefoghe, C and Debille, J. Advanced system identification methods for improved service load simulation on multi axial test rigs. European Journal of Mechanical and Environmental Engineering; 1999:44(1):27-39.

[28] Deckers, K, Guillaume, P, Lefebvre, D, De Baere, D. Turning point based fatigue testing: combining multisines with turning point replication. Mechanical Systems and Signal Processing; 2012:30:23-31.

[29] Eksteen, J, Raath, A. Time-domain reconstruction of dynamic multiaxial responses in aeronautical fatigue testing. Journal of Aircraft; 2001:38(1):147-153.

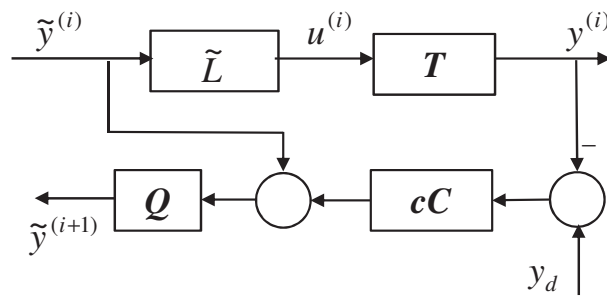


Figure 3. Modified version of the alternative algorithm for ILC on a nonlinear system T with a nonlinear L .

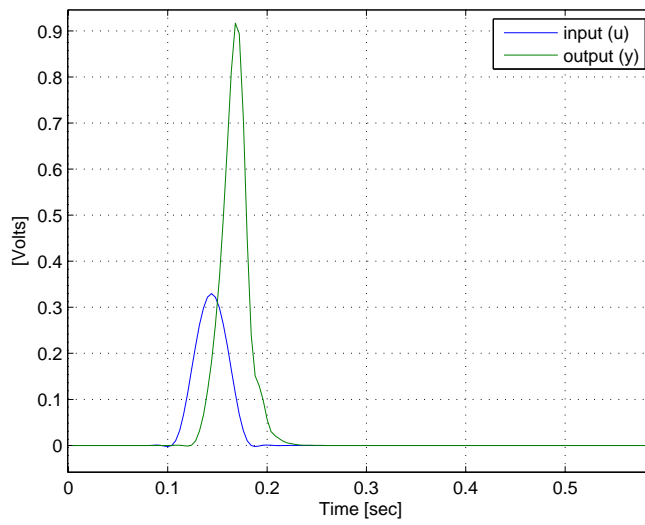


Figure 4. Example 1: Desired input $u_d(t)$ and output $y_d(t)$ data.

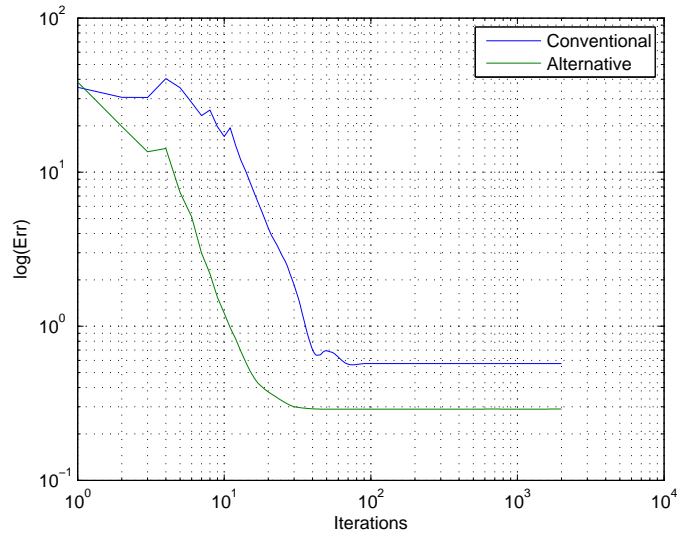


Figure 5. Example 1: Convergence error of the input with respect to the desired input for the conventional and alternative ILC algorithms.

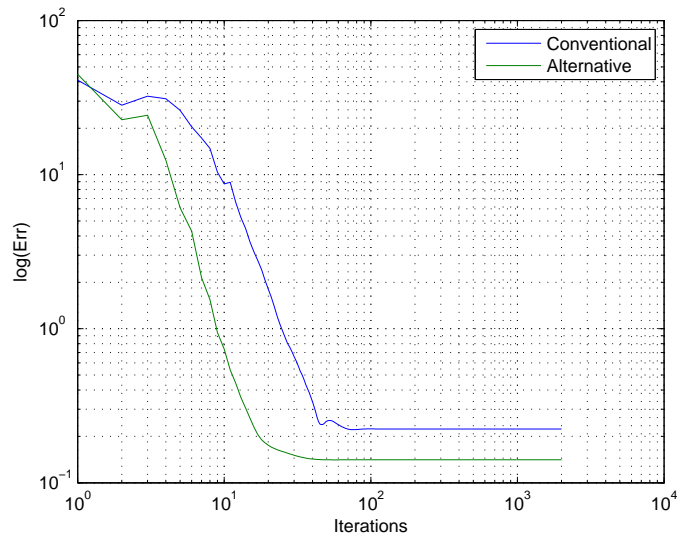


Figure 6. Example 1: Convergence error of the output with respect to the desired response for the conventional and alternative ILC algorithms.

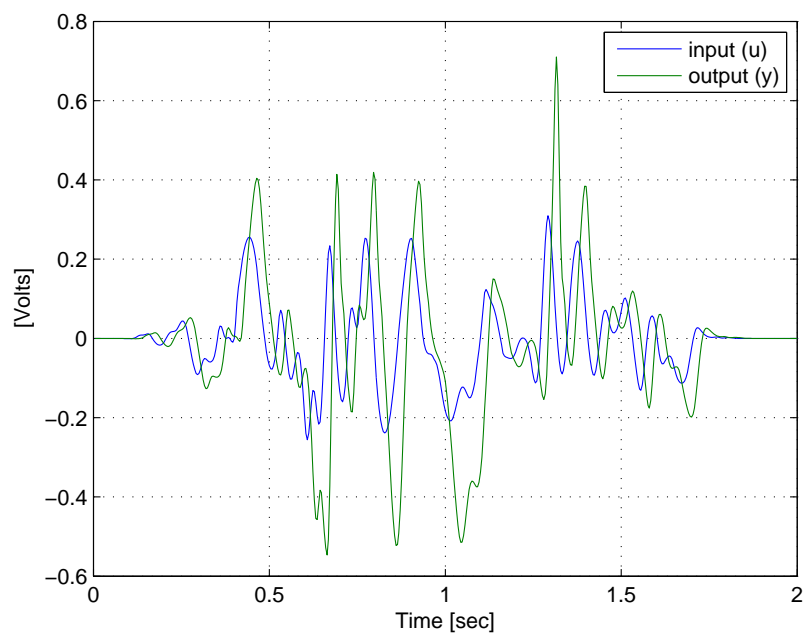


Figure 7. Example 2: Desired input $u_d(t)$ and response $y_d(t)$ data.

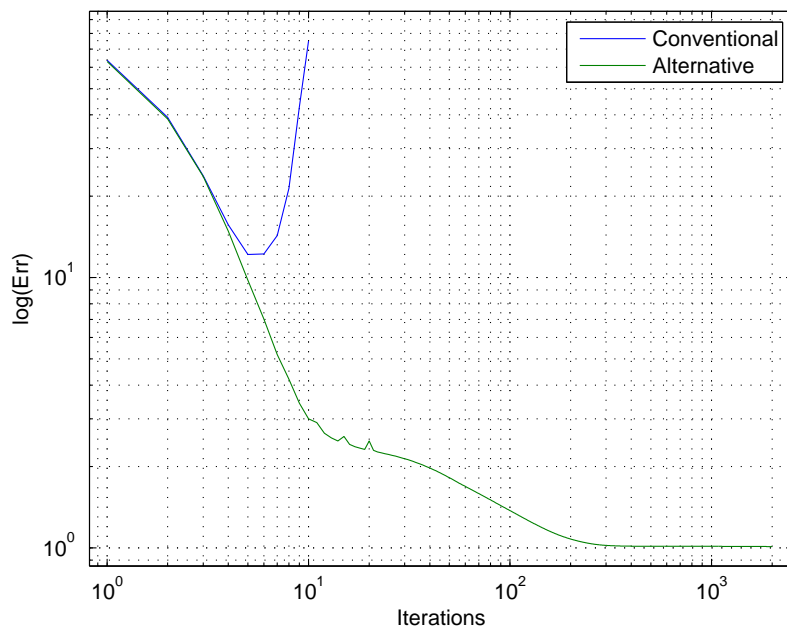


Figure 8. Example 2: Convergence error of the input with respect to the conventional and alternative ILC algorithms.

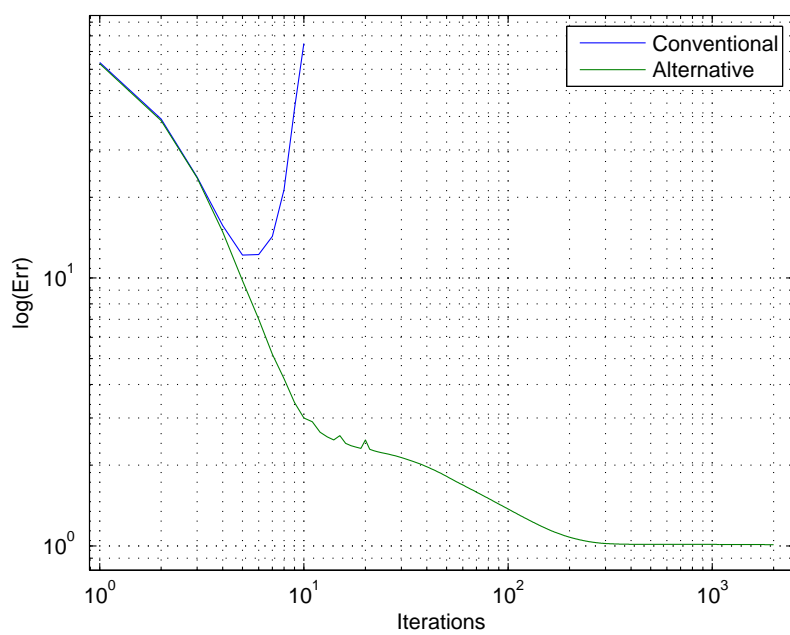


Figure 9. Example 2: Convergence error of the output with respect to the conventional and alternative ILC algorithms.

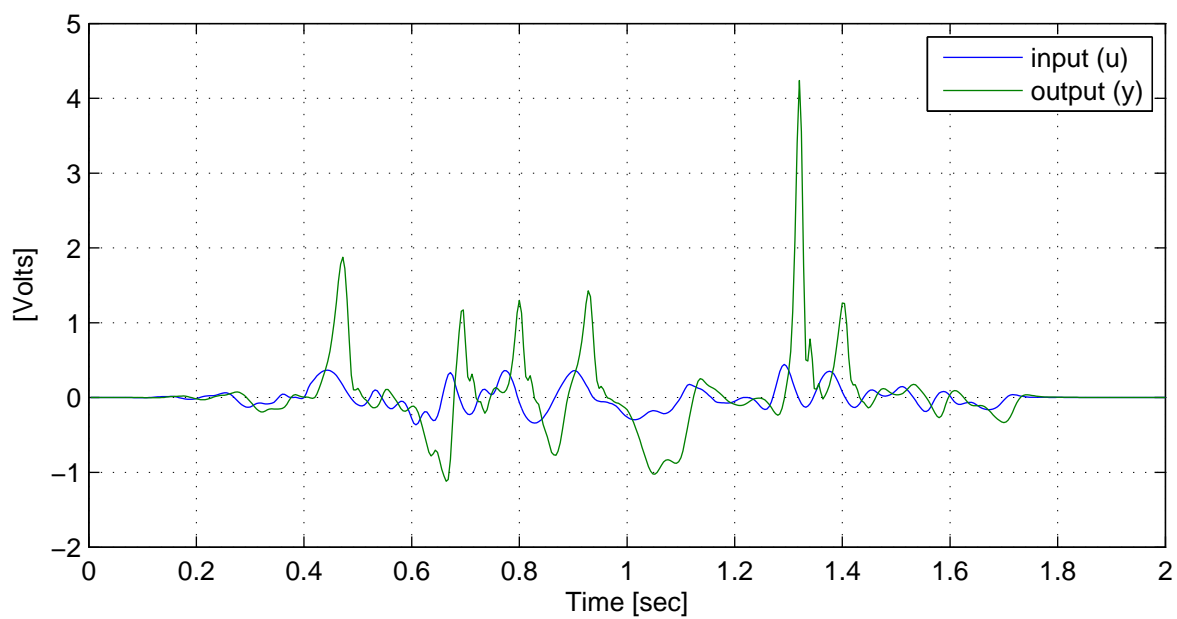


Figure 10. Example 3: Desired input $u_d(t)$ and output $y_d(t)$ data.