

# A statistical analysis of certain iterative learning control algorithms

M. BUTCHER<sup>\*\*</sup>; A. KARIMI and R. LONGCHAMP  
Automatic Control Laboratory, Ecole Polytechnique Fédéral de  
Lausanne (EPFL), 1015 Lausanne, Switzerland

This is a preprint of an article whose final and definitive form  
will be published in the INTERNATIONAL JOURNAL OF  
CONTROL [2007] [copyright Taylor & Francis];  
INTERNATIONAL JOURNAL OF CONTROL is available  
online at: <http://journalsonline.tandf.co.uk>;

September 21, 2007

## Abstract

Iterative Learning Control (ILC) is a technique used to improve the tracking performance of systems carrying out repetitive tasks, which are affected by deterministic disturbances. The achievable performance is greatly degraded, however, when non-repeating, stochastic disturbances are present. This paper aims to compare a number of different ILC algorithms, proposed to be more robust to the presence of these disturbances, firstly by a statistical analysis and then by simulation results and their application to a linear motor. New expressions for the expected value and variance of the controlled error are developed for each algorithm. The different algorithms are then tested in simulation and finally applied to the linear motor system to test their performance in practice. A filtered ILC algorithm is proposed when the noise and desired output spectra are separated. Otherwise an algorithm with a decreasing gain gives good robustness to noise and achievable precision but at a slower convergence rate.

## 1 INTRODUCTION

Iterative Learning Control (ILC) is a technique used to improve the tracking performance of systems that carry out repetitive tasks. Based on the process of human learning, information ‘learnt’ from the previous repetitions is used to improve the performance of the system during the next repetition/iteration i.e. reduce the tracking error. Since the first publications on ILC e.g. [1] the subject has been intensely researched producing a number of surveys [2, 3] and books [4, 5], as well as numerous papers. The technique has been demonstrated to be capable of considerably improving the tracking performance of systems that are predominately affected by deterministic disturbances.

---

<sup>\*\*</sup>Corresponding author: [mark.butcher@epfl.ch](mailto:mark.butcher@epfl.ch)

Because these are repeated from one repetition to the next, ILC is capable of learning them and, in turn, compensating for them. However, when the system is affected by stochastic, non-repeating disturbances, such as measurement noise, the achievable performance is greatly degraded. It is, therefore, necessary to understand exactly how stochastic disturbances affect the learning process and to develop ILC algorithms that are robust to their presence.

Some research has already been done into the influence of disturbances on ILC [6], although it has not been as widely investigated as the deterministic aspects. In [7] a disturbance analysis is done and both recursive and explicit expressions for the measured error in terms of desired output and disturbances are found. They are used to discuss generally how the presence of disturbances affects the measured error evolution. In [8] a similar analysis is done and simulation and experimental results are used to illustrate the general effect of iteration-dependent disturbances.

Recognising the degrading effect of stochastic iteration-dependent disturbances, certain researchers have proposed algorithms which are less sensitive to their presence.

1. The use of a forgetting factor in ILC was first proposed in [9] for a D-type ILC law. It was then proposed in [10] for P-type ILC. The underlying idea is that as the iterations progress, information from the oldest iterations will be ‘forgotten’ and thus the stochastic disturbances from these will have less effect on the current input.
2. In [11] another ILC algorithm is proposed using a learning gain that decreases each iteration and has the form of a Stochastic Approximation algorithm. An iteration decreasing learning gain is also used in [12] for repetitive disturbance rejection in the presence of measurement noise. Iteration varying filters were also used in a stochastic ILC law in [13] and an adaptive ILC law in [14].
3. The filtering of the ILC command has been proposed in certain papers, as well, as a way of reducing the influence of noise on the error [6].

The aim of this paper is to compare these different ILC algorithms firstly by a statistical analysis, then in simulation and finally by their application to a linear motor system. A recursive formulation for the controlled error will be used to develop a new transfer function relationship in the iteration domain. This then allows novel expressions for both the expected value and variance of the error for each algorithm to be developed.

Recently in [15] analytical expressions for the covariance matrix of the controlled error were developed for high order ILC algorithms with load and measurement disturbances separately. These expressions together are similar to the complete variance expression derived in this paper, though here transfer operators are used instead of matrix operators allowing certain insightful frequency domain expressions to be derived. The contributions differ, also, as here a more complete analysis is made including expressions for the expected value of the controlled error and analytical and experimental comparisons of specific algorithms are given.

This paper is organised as follows. In Section 2 the ILC problem is formulated and an insight is given into how iteration-varying disturbances affect the error convergence. Additionally, certain notations are defined and the assumptions used during the analysis are given. Then in Section 3 expressions for the expected value and variance of the error are developed. In Section 4 the expressions are used to analyse the different ILC algorithms. Section 5 tests the algorithms, first in simulation, then experimentally on the linear motor system. Finally in Section 6 some conclusions are made.

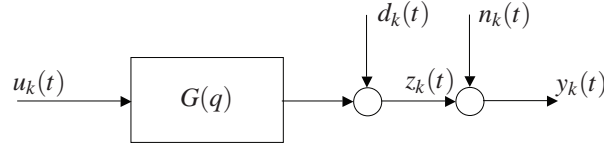


Figure 1: System with a repetitive desired output

## 2 PRELIMINARIES AND NOTATIONS

We consider the SISO, linear, discrete-time system  $G(q)$ , shown in Fig. 1, whose controlled output,  $z_k(t)$ , at repetition  $k$  is given by:

$$z_k(t) = G(q)u_k(t) + d_k(t) \quad (1)$$

where  $u_k(t)$  is the input to the system,  $d_k(t)$  the iteration-varying load disturbance and  $q$  the forward-shift, time domain operator. Additionally the measured output is given by:

$$y_k(t) = z_k(t) + n_k(t) \quad (2)$$

where  $n_k(t)$  is the iteration-varying measurement disturbance.

A general form of an ILC command is given by:

$$u_{k+1}(t) = Q(q)(u_k(t) + L(q)e_k(t)) \quad (3)$$

where  $Q(q)$  and  $L(q)$  are linear, discrete and possibly non-causal filters and the measured error signal is given by:

$$e_k(t) = y_d(t) - y_k(t) \quad (4)$$

where  $y_d(t)$  is the desired system output, defined over the repetition duration for  $t = 0, \dots, N-1$ . To analyse the tracking ability of an ILC algorithm it is of more interest to analyse an error signal based on the controlled output, rather than the measured one, so the controlled error signal is defined as:

$$\varepsilon_k(t) = y_d(t) - z_k(t). \quad (5)$$

From hereon  $\varepsilon_k(t)$  will simply be referred to as the error rather than the controlled error. By combining equations (1) - (5) a recursive expression for the controlled error evolution equation is found to be:

$$\begin{aligned} \varepsilon_{k+1}(t) = & Q(q)[1 - G(q)L(q)]\varepsilon_k(t) \\ & + [1 - Q(q)]y_d(t) + Q(q)d_k(t) - d_{k+1}(t) \\ & + G(q)Q(q)L(q)n_k(t). \end{aligned} \quad (6)$$

It is clear that even if  $L(q)$  is chosen so that  $L(q) = G^{-1}(q)$  the presence of iteration-varying disturbances, and the use of a filter  $Q(q) \neq 1$ , mean that a steady error value equal to zero is not achievable.

In the absence of disturbances it can be seen that the quickest convergence of the error can be achieved by taking

$$L(q) = G^{-1}(q), \quad (7)$$

however finding even a good approximation of  $G^{-1}(q)$  is a laborious task, and  $G^{-1}(q)$  exactly impossible. A frequently encountered, sufficient condition for monotonic convergence of the error between iterations in the 2-norm, when either both  $Q(q)$  and  $L(q)$  are causal or  $N = \infty$ , is the following, e.g. [16]:

$$\max_{\omega \in [0, \omega_N]} \left| Q(e^{j\omega h}) \left( 1 - G(e^{j\omega h}) L(e^{j\omega h}) \right) \right| < 1 \quad (8)$$

where  $\omega_N$  is the Nyquist frequency and  $h$  the sampling period. Therefore, often  $L(q)$  is chosen so as to satisfy this condition and can be expressed generally as:

$$\begin{aligned} L(q) &= \hat{G}^{-1}(q) \\ &= (1 + \Delta(q)) G^{-1}(q), \end{aligned} \quad (9)$$

where  $\hat{G}(q)$  is a model of the system and  $\Delta(q)$  represents the multiplicative uncertainty due to the unmodelled dynamics i.e.

$$G(q) = [1 + \Delta(q)] \hat{G}(q). \quad (10)$$

It should be noted that the direct use of the model inverse is only one choice for  $L(q)$  from a number of approaches proposed in the literature. It is used here, however, because many of these other choices can be seen as approximations to the inverse model, each with their own associated  $\Delta(q)$ .

In [17] a forward shift operator,  $w$ , is defined, which works in the iteration domain. It has the property:

$$v_{k+1}(t) = wv_k(t) \quad (11)$$

where  $v(t)$  is an arbitrary variable. Using this operator it is possible to rewrite equation (6) as:

$$\begin{aligned} \varepsilon_k(t) &= \frac{[1 - Q(q)]y_d(t) + [Q(q) - w]d_k(t) + G(q)Q(q)L(q)n_k(t)}{w - Q(q)[1 - G(q)L(q)]}. \end{aligned} \quad (12)$$

This expression will be used later to analyse the effect of disturbances.

## 2.1 Assumptions

In order to make a detailed analysis of the effect of the disturbances on ILC algorithms, it is necessary to first make some assumptions about the nature of the disturbances. It is possible that the load and measurement disturbances can be either iteration repetitive or iteration-varying. Repetitive disturbances will be learnt gradually, in a similar way to the reference signal, and thus will not be considered here. The following assumptions are thus made:

**(A1)** The load disturbance  $d_k(t)$  and measurement disturbance  $n_k(t)$  are iteration-varying.

**(A2)**  $d_k(t)$  and  $n_k(t)$  are taken as being zero mean, weakly stationary sequences, which are white in the iteration-domain with variances equal to  $\sigma_d^2$  and  $\sigma_n^2$  respectively i.e.  $E\{d_k(t)\} = 0$  and  $E\{n_k(t)\} = 0$ ,

$$E\{d_k(t)d_{k+m}(t)\} = \begin{cases} \sigma_d^2 & m = 0 \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

and

$$E\{n_k(t)n_{k+m}(t)\} = \begin{cases} \sigma_n^2 & m = 0 \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

where  $E\{\cdot\}$  denotes the mathematical expectation and  $m$  the lag in the iteration-domain.

**(A3)**  $d_k(t)$  and  $n_k(t)$  are assumed to be uncorrelated with the system input  $u_k(t)$  and with each other i.e.

$$E\{u_k(t)d_k(t+m)\} = 0 \quad \forall m, \quad (15)$$

$$E\{u_k(t)n_k(t+m)\} = 0 \quad \forall m, \quad (16)$$

$$E\{d_k(t)n_k(t+m)\} = 0 \quad \forall m. \quad (17)$$

### 3 EXPECTED VALUE AND VARIANCE EXPRESSIONS

It is desired to calculate the statistical properties of an ILC algorithm in order to see how the presence of noise affects the error achieved. The main results are given below in the form of theorems.

**Theorem 1** *For the system described by equations (1)-(2), using the ILC algorithm (3), and respecting the assumptions (A1)-(A3), the converged expected value of the error signal is:*

$$\lim_{k \rightarrow \infty} E\{\varepsilon_k(t)\} = \frac{[1 - Q(q)]}{1 - Q(q)[1 - G(q)L(q)]} y_d(t). \quad (18)$$

**Proof 1** *Using equation (12) we find:*

$$\begin{aligned} & E\{\varepsilon_k(t)\} \\ = & E\left\{ \frac{[1 - Q(q)]y_d(t) + [Q(q) - w]d_k(t) + G(q)Q(q)L(q)n_k(t)}{w - Q(q)[1 - G(q)L(q)]} \right\} \\ = & \frac{1}{w - Q(q)[1 - G(q)L(q)]} ([1 - Q(q)]E\{y_d(t)\} \\ & + [Q(q) - w]E\{d_k(t)\} + G(q)Q(q)L(q)E\{n_k(t)\}) \\ = & \frac{[1 - Q(q)]}{w - Q(q)[1 - G(q)L(q)]} y_d(t). \end{aligned} \quad (19)$$

The  $z$ -transform of equation (19) in the iteration domain can be taken:

$$\mathcal{Z}\{E\{\varepsilon_k(t)\}\} = \frac{[1 - Q(q)]}{z - Q(q)[1 - G(q)L(q)]} \cdot \frac{y_d(t)z}{z - 1} \quad (20)$$

where  $y_d(t)$ , being constant from one iteration to the next, acts as a step input at iteration  $k = 0$ . In order to find the value of the error signal as the iteration number tends to infinity it is possible to use the standard Final Value Theorem for discrete systems,

under the assumption that condition (8) is satisfied and thus the system poles in the iteration domain are within the unit circle. This gives:

$$\begin{aligned} & \lim_{k \rightarrow \infty} E\{\varepsilon_k(t)\} \\ &= \lim_{z \rightarrow 1} \frac{(z-1)[1-Q(q)]}{z-Q(q)[1-G(q)L(q)]} \cdot \frac{y_d(t)z}{z-1} \\ &= \frac{[1-Q(q)]}{1-Q(q)[1-G(q)L(q)]} y_d(t). \end{aligned}$$

**Theorem 2** For the system described by equations (1)-(2), using the ILC algorithm (3), and respecting the assumptions (A1)-(A3), the variance of the error signal at iteration  $k+1$  is:

$$\begin{aligned} E\{\tilde{\varepsilon}_{k+1}^2(t)\} &= E\{[B(q)\tilde{\varepsilon}_k(t)]^2\} + E\{[Q(q)d_k(t)]^2\} + E\{d_{k+1}^2(t)\} \\ &\quad + E\{[G(q)Q(q)L(q)n_k(t)]^2\} - 2E\{B(q)d_k(t)Q(q)d_k(t)\}, \end{aligned} \quad (21)$$

where  $B(q) = Q(q)[1-G(q)L(q)]$ .

**Proof 2** The variance of the error at iteration  $k$  is defined as:

$$E\{\tilde{\varepsilon}_k^2(t)\} = E\{[\varepsilon_k(t) - E\{\varepsilon_k(t)\}]^2\}. \quad (22)$$

Using equations (12) and (19) we find:

$$\tilde{\varepsilon}_k(t) = \frac{[Q(q) - w]d_k(t) + G(q)Q(q)L(q)n_k(t)}{w - B(q)} \quad (23)$$

Expanding equation (23) into its recursive form in the iteration domain gives:

$$\tilde{\varepsilon}_{k+1}(t) = B\tilde{\varepsilon}_k(t) + Qd_k(t) - d_{k+1}(t) + GQLn_k(t), \quad (24)$$

where  $q$  has been omitted for simplicity. Taking the square of equation (24) gives:

$$\begin{aligned} \tilde{\varepsilon}_{k+1}^2(t) &= [B\tilde{\varepsilon}_k(t)]^2 + [Qd_k(t)]^2 + d_{k+1}^2(t) + [GQLn_k(t)]^2 \\ &\quad + 2[B\tilde{\varepsilon}_k(t)Qd_k(t) - d_{k+1}(t)B\tilde{\varepsilon}_k(t) \\ &\quad + B\tilde{\varepsilon}_k(t)GQLn_k(t) - d_{k+1}(t)Qd_k(t) \\ &\quad + Qd_k(t)GQLn_k(t) - d_{k+1}(t)GQLn_k(t)], \end{aligned} \quad (25)$$

which on applying the expectation operator leads to:

$$\begin{aligned} E\{\tilde{\varepsilon}_{k+1}^2(t)\} &= E\{[B\tilde{\varepsilon}_k(t)]^2\} + E\{[Qd_k(t)]^2\} + E\{d_{k+1}^2(t)\} \\ &\quad + E\{[GQLn_k(t)]^2\} + 2E\{B\tilde{\varepsilon}_k(t)Qd_k(t)\}, \end{aligned} \quad (26)$$

where many cross terms are lost as they are uncorrelated. The remaining cross term, the last term in equation (26), can be found by filtering equation (24), evaluated at iteration  $k$ , by  $B(q)$  and then multiplying by  $Q(q)d_k(t)$  and taking the expected value, this gives:

$$\begin{aligned} E\{B\tilde{\varepsilon}_k(t)Qd_k(t)\} &= E\{B^2\tilde{\varepsilon}_{k-1}(t)Qd_k(t)\} \\ &\quad + E\{BQd_{k-1}(t)Qd_k(t)\} - E\{Bd_k(t)Qd_k(t)\} \\ &\quad + E\{BGQLn_{k-1}(t)Qd_k(t)\} \\ &= -E\{Bd_k(t)Qd_k(t)\}, \end{aligned} \quad (27)$$

which on substitution into equation (26) gives:

$$E\{\tilde{\varepsilon}_{k+1}^2(t)\} = E\{[B\tilde{\varepsilon}_k(t)]^2\} + E\{[Qd_k(t)]^2\} + E\{d_{k+1}^2(t)\} \\ + E\{[GQLn_k(t)]^2\} - 2E\{Bd_k(t)Qd_k(t)\}, \quad (28)$$

which leads to the expression in Theorem 2.

**Remarks:**

1. It is clear from Theorem 1, as remarked earlier, that the use of a filter  $Q(q) \neq 1$  does not allow a zero error value to be reached.
2. Using Theorems 1 and 2 a value for the 2-norm of the converged error can be found. Using the fact that the variance can be expressed as:

$$E\{\tilde{\varepsilon}_k^2(t)\} = E\{\varepsilon_k^2(t)\} - E\{\varepsilon_k(t)\}^2, \quad (29)$$

we get that:

$$E\{\varepsilon_k^2(t)\} = E\{\tilde{\varepsilon}_k^2(t)\} + E\{\varepsilon_k(t)\}^2. \quad (30)$$

The 2-norm of the converged error is then defined as:

$$\|\varepsilon_k\|_2 = \left( \frac{1}{N} \sum_{t=0}^{N-1} E\{\varepsilon_k^2(t)\} \right)^{1/2} \\ = \left( \frac{1}{N} \sum_{t=0}^{N-1} [E\{\tilde{\varepsilon}_k^2(t)\} + E\{\varepsilon_k(t)\}^2] \right)^{1/2}. \quad (31)$$

This illustrates that a small 2-norm value is only achievable when both the variance and the expected value of the error are small, thus motivating the need for an analysis of these quantities.

## 4 ANALYSIS OF ALGORITHMS

In this section various previously proposed algorithms shall be analysed using the expressions developed in the previous section.

### 4.1 Non disturbance-robust algorithm

In the standard case where the ILC law is designed without consideration for stochastic disturbances, we have  $Q(q) = 1$  and  $L(q) = [1 + \Delta(q)]G^{-1}(q)$ . With these values Theorem 1 gives

$$\lim_{k \rightarrow \infty} E\{\varepsilon_k(t)\} = 0 \quad (32)$$

and Theorem 2 gives:

$$E\{\tilde{\varepsilon}_{k+1}^2(t)\} = E\{[\Delta(q)\tilde{\varepsilon}_k(t)]^2\} + E\{d_k^2(t)\} + E\{d_{k+1}^2(t)\} \\ + E\{[(1 + \Delta(q))n_k(t)]^2\} + 2E\{d_k(t)\Delta(q)d_k(t)\}. \quad (33)$$

Thus the expected value of the error can be seen to eventually converge to 0, as desired. However, in the case of perfect knowledge of the system i.e.  $\Delta(q) = 0$ , which is wanted for rapid deterministic convergence, the variance of the error signal is:

$$E\{\tilde{\epsilon}_{k+1}^2(t)\} = E\{d_k^2(t)\} + E\{d_{k+1}^2(t)\} + E\{n_k^2(t)\} = 2\sigma_d^2 + \sigma_n^2, \quad (34)$$

where  $\sigma_d^2$  is the variance of both  $d_k(t)$  and  $d_{k+1}(t)$  due to the stationarity assumption made on  $d_k(t)$ . Thus the variance of the error is the sum of twice the variance of the load disturbance and the variance of the measurement disturbance. The fact that the load disturbance's variance is doubled corresponds to a result found in [11], though there only a single perturbation source is considered, and demonstrates how the presence of non-repetitive disturbances can be particularly detrimental to the tracking performance achievable using ILC as they are fed back from the previous iteration.

**Remark:** If, instead of taking  $L(q)$  as above, it is taken as  $L(q) = \beta[1 + \Delta(q)]G^{-1}(q)$ , with  $Q(q) = 1$  still, the expected value of the error still converges to zero in the limit, provided the convergence condition (8) is fulfilled, but the variance of the error is now:

$$\begin{aligned} E\{\tilde{\epsilon}_{k+1}^2(t)\} = & E\{[(1 - \beta[1 + \Delta(q)])\tilde{\epsilon}_k(t)]^2\} + E\{d_k^2(t)\} \\ & + E\{d_{k+1}^2(t)\} + E\{[\beta[1 + \Delta(q)]n_k(t)]^2\} \\ & - 2E\{d_k(t)(1 - \beta[1 + \Delta(q)])d_k(t)\}, \end{aligned} \quad (35)$$

which in the case of perfect system knowledge leads to:

$$\begin{aligned} E\{\tilde{\epsilon}_{k+1}^2(t)\} = & (1 - \beta)^2 E\{\tilde{\epsilon}_k^2(t)\} - (1 - 2\beta)E\{d_k^2(t)\} \\ & + E\{d_{k+1}^2(t)\} + \beta^2 E\{n_k^2(t)\}. \end{aligned} \quad (36)$$

Since  $0 < \beta < 1$ ,  $(1 - \beta)^2 < 1$  and so  $\lim_{k \rightarrow \infty} E\{\tilde{\epsilon}_{k+1}^2(t)\} = E\{\tilde{\epsilon}_k^2(t)\} = E\{\tilde{\epsilon}_\infty^2(t)\}$ , where  $E\{\tilde{\epsilon}_\infty^2(t)\}$  is given by:

$$E\{\tilde{\epsilon}_\infty^2(t)\} = \frac{2}{(2 - \beta)}\sigma_d^2 + \frac{\beta}{(2 - \beta)}\sigma_n^2. \quad (37)$$

So the limit variance of the error can be reduced compared to that which occurs in the standard case, when  $\beta = 1$ . This reduction in the error variance entails, nonetheless, a reduction in the rate of convergence of the deterministic error and thus a compromise must be made. The expression for the component of the error variance due to the load disturbances is in agreement with an expression found in Section 6.2 of [15].

## 4.2 The use of a forgetting factor

An ILC law with a forgetting factor is given by:

$$u_{k+1}(t) = (1 - \alpha)u_k(t) + L'(q)e_k(t), \quad (38)$$

where  $0 \leq \alpha < 1$  is the forgetting factor. The objective of introducing the forgetting factor is to increase the learning algorithm's robustness to initialisation errors, fluctuations of the dynamics and random disturbances. The thinking behind this is that when considering the input at the  $k$ 'th iteration the previous inputs will be multiplied by  $(1 - \alpha)^k$ . Thus when  $\alpha$  is chosen such that  $1 - \alpha < 1$  their influence on the current input should be diminished, and so will that of the disturbances from previous iterations that are fed back in the inputs. The law (38) will now be investigated using the framework presented above.



By comparing the ILC laws (3) and (38), it is possible to see that  $Q(q) = 1 - \alpha$  and  $L(q) = \frac{L'(q)}{1-\alpha}$ . If  $L'(q) = [1 + \Delta(q)]G^{-1}(q)$  is taken, Theorem 1 gives:

$$\lim_{k \rightarrow \infty} E\{\varepsilon_k(t)\} = \frac{\alpha}{\alpha + [1 + \Delta(q)]} y_d(t). \quad (39)$$

So when  $\alpha \neq 0$  the expected value of the error cannot converge to zero. Examining the variance, Theorem 2 gives:

$$\begin{aligned} E\{\tilde{\varepsilon}_{k+1}^2(t)\} &= E\{[(\alpha + \Delta(q))\tilde{\varepsilon}_k(t)]^2\} + (1 - \alpha)^2 E\{d_k^2(t)\} \\ &\quad + E\{d_{k+1}^2(t)\} + E\{[(1 + \Delta(q))n_k(t)]^2\} \\ &\quad + 2(1 - \alpha)E\{d_k(t)[\alpha + \Delta(q)]d_k(t)\}. \end{aligned} \quad (40)$$

If we have perfect knowledge of the system,  $\Delta(q) = 0$ , then, in the limit  $k \rightarrow \infty$ , equation (40) reduces to:

$$\begin{aligned} E\{\tilde{\varepsilon}_\infty^2(t)\} &= \alpha^2 E\{\tilde{\varepsilon}_\infty^2(t)\} + (1 - \alpha)^2 E\{d_k^2(t)\} \\ &\quad + E\{d_{k+1}^2(t)\} + E\{n_k^2(t)\} \\ &\quad + 2\alpha(1 - \alpha)E\{d_k^2(t)\} \\ \Rightarrow E\{\tilde{\varepsilon}_\infty^2(t)\} &= \sigma_d^2 + \frac{1}{1 - \alpha^2}(\sigma_d^2 + \sigma_n^2). \end{aligned} \quad (41)$$

Thus the forgetting factor that minimises the limit variance of the error can be seen to be  $\alpha = 0$  i.e. no forgetting factor should be used, giving the standard algorithm. This result is similar to a result found in [18], which uses a different analysis framework but concludes that the best forgetting matrix is zero when the trace of the input error covariance matrix is minimised.

It is interesting to note that if  $L'(q)$  is chosen according to equation (9),  $L'(q) = L(q) = [1 + \Delta(q)]G^{-1}(q)$ , and  $Q(q)$  is as before i.e. the forgetting factor affects the entire algorithm, a different conclusion is reached. In this case Theorem 1 gives:

$$\lim_{k \rightarrow \infty} E\{\varepsilon_k(t)\} = \frac{\alpha}{1 + (1 - \alpha)\Delta(q)} y_d(t). \quad (42)$$

So again we see that in order to minimise  $\lim_{k \rightarrow \infty} E\{\varepsilon_k(t)\}$  the optimal  $\alpha$  is  $\alpha = 0$ . But Theorem 2 gives:

$$\begin{aligned} E\{\tilde{\varepsilon}_{k+1}^2(t)\} &= (1 - \alpha)^2 E\{[\Delta(q)\tilde{\varepsilon}_k(t)]^2\} + (1 - \alpha)^2 E\{d_k^2(t)\} \\ &\quad + E\{d_{k+1}^2(t)\} + (1 - \alpha)^2 E\{[(1 + \Delta(q))n_k(t)]^2\} \\ &\quad + 2(1 - \alpha)^2 E\{d_k(t)\Delta(q)d_k(t)\}, \end{aligned} \quad (43)$$

which on assuming perfect system knowledge gives:

$$\begin{aligned} E\{\tilde{\varepsilon}_{k+1}^2(t)\} &= E\{d_{k+1}^2(t)\} + (1 - \alpha)^2 (E\{d_k^2(t)\} + E\{n_k^2(t)\}) \\ &= \sigma_d^2 + (1 - \alpha)^2 (\sigma_d^2 + \sigma_n^2). \end{aligned} \quad (44)$$

Thus this time the optimal value of  $\alpha$  to minimise the variance of the error is  $\alpha = 1$  and leads to  $E\{\tilde{\varepsilon}_{k+1}^2(t)\} = \sigma_d^2$ . This value makes sense as it means that the previous input is not fed back at all so only the load disturbances during the current iteration affect the error. A compromise therefore needs to be made between minimising the variance of the error and keeping its converged, expected value small.

### 4.3 The use of an iteration decreasing learning gain

An ILC command with an iteration decreasing learning gain has the form:

$$u_{k+1}(t) = u_k(t) + \frac{L'(q)}{k+1} e_k(t). \quad (45)$$

With this law  $Q(q) = 1$  and  $L(q) = \frac{L'(q)}{k+1}$  thus Theorem 1 gives:

$$\lim_{k \rightarrow \infty} E\{\varepsilon_k(t)\} = 0, \quad (46)$$

and Theorem 2, with the assumption of perfect system knowledge:

$$\begin{aligned} E\{\tilde{\varepsilon}_{k+1}^2(t)\} &= E\left\{\left[\left(1 - \frac{1}{k+1}\right)\tilde{\varepsilon}_k(t)\right]^2\right\} + E\{d_k^2(t)\} \\ &\quad + E\{d_{k+1}^2(t)\} + E\left\{\frac{n_k^2(t)}{(k+1)^2}\right\} \\ &\quad - 2E\left\{d_k(t)\left(1 - \frac{1}{k+1}\right)d_k(t)\right\} \\ &= \left(1 - \frac{1}{k+1}\right)^2 E\{\tilde{\varepsilon}_k^2(t)\} + \frac{1}{(k+1)^2} E\{n_k^2(t)\} \\ &\quad + \frac{2}{k+1} E\{d_k^2(t)\}. \end{aligned} \quad (47)$$

It can be seen that when  $k \rightarrow \infty$   $E\{\tilde{\varepsilon}_{k+1}^2(t)\} = E\{\tilde{\varepsilon}_k^2(t)\} = E\{\tilde{\varepsilon}_\infty^2(t)\}$  so we have:

$$\begin{aligned} E\{\tilde{\varepsilon}_\infty^2(t)\} &= \left(1 - \frac{2}{k+1} + \frac{1}{(k+1)^2}\right) E\{\tilde{\varepsilon}_\infty^2(t)\} \\ &\quad + \frac{1}{(k+1)^2} E\{n_k^2(t)\} + \frac{2}{k+1} E\{d_k^2(t)\} \\ \Rightarrow E\{\tilde{\varepsilon}_\infty^2(t)\} &= \frac{1}{2(k+1)} [E\{\tilde{\varepsilon}_\infty^2(t)\} + E\{n_k^2(t)\}] + E\{d_k^2(t)\}, \end{aligned} \quad (48)$$

which gives in the limit:

$$\lim_{k \rightarrow \infty} E\{\tilde{\varepsilon}_\infty^2(t)\} = E\{d_k^2(t)\} = \sigma_d^2. \quad (49)$$

This shows that using a decreasing learning gain asymptotically the expected value of the error converges to zero and the variance of the error becomes equal to just that of the load disturbance. This is the best that can be achieved. The disadvantage of this algorithm is that the error contraction rate reduces with iteration number and eventually the learning practically stops so it cannot react to changes in the desired output or the repetitive disturbances affecting the system.

### 4.4 The use of a filter

A filtered ILC law is that given by the general form (3) i.e.  $u_{k+1}(t) = Q(q)[u_k(t) + L(q)e_k(t)]$ , where  $Q(q)$  is the filter referred to. When  $L(q) = [1 + \Delta(q)]G^{-1}(q)$  is used, Theorem 1 gives:

$$\lim_{k \rightarrow \infty} E\{\varepsilon_k(t)\} = \frac{[1 - Q(q)]}{1 + Q(q)\Delta(q)} y_d(t). \quad (50)$$

ILC is, by nature, defined over a finite time duration,  $N$ . This means that the infinite-time Fourier transform cannot be calculated in order to work in the frequency domain. However, when  $N$  is large compared to the settling time of  $G(q)$  the finite-time Fourier transform can be used to make a reasonably accurate frequency domain analysis. The magnitude response of the Fourier transform of equation (50) is:

$$\left| \mathcal{F} \left\{ \lim_{k \rightarrow \infty} E \{ \varepsilon_k(t) \} \right\} \right|^2 = \frac{|1 - Q(e^{j\omega h})|^2}{|1 + Q(e^{j\omega h})\Delta(e^{j\omega h})|^2} |Y_d(\omega)|^2.$$

It is clear that in order to converge to a zero expected error it is necessary to use a filter whose magnitude is equal to 1, and has zero phase shift, at frequencies where  $Y_d(\omega)$  has non-zero components.

Now considering the error variance, Theorem 2 gives:

$$\begin{aligned} E \{ \tilde{\varepsilon}_{k+1}^2(t) \} &= E \{ [Q(q)\Delta(q)\tilde{\varepsilon}_k(t)]^2 \} + E \{ [Q(q)d_k(t)]^2 \} \\ &\quad + E \{ d_{k+1}^2(t) \} + E \{ [Q(q)[1 + \Delta(q)]n_k(t)]^2 \} \\ &\quad + 2E \{ Q(q)\Delta(q)d_k(t)Q(q)d_k(t) \} \end{aligned} \quad (51)$$

Again examining this for the specific case of perfect system knowledge the above expression becomes:

$$E \{ \tilde{\varepsilon}_{k+1}^2(t) \} = \sigma_d^2 + E \{ [Q(q)d_k(t)]^2 \} + E \{ [Q(q)n_k(t)]^2 \}. \quad (52)$$

Since  $d_k(t)$  and  $n_k(t)$  are stationary so is  $\varepsilon_{k+1}(t)$ , therefore (52) can also be expressed as:

$$E \{ \tilde{\varepsilon}_{k+1}^2(t) \} = \sigma_d^2 + \frac{h}{2\pi} \int_{-\pi/h}^{\pi/h} |Q(e^{j\omega h})|^2 [\Phi_d(\omega) + \Phi_n(\omega)] d\omega \quad (53)$$

where  $\Phi_d(\omega)$  and  $\Phi_n(\omega)$  are the power spectra of  $d_k(t)$  and  $n_k(t)$ , respectively. It is thus possible to see that the variance of the error can be reduced below that obtained using the standard algorithm by choosing  $Q(q)$  to have a magnitude of less than one at frequencies at which the disturbance power spectra are large.

A compromise, therefore, needs to be made again, between filtering in order to reduce the error variance, but not filtering at frequencies important to  $y_d(t)$  so to allow a reasonable converged error to be achieved. Fortunately it is normal to use low-frequency signals for  $y_d(t)$ , whilst  $\Phi_d(\omega)$  and  $\Phi_n(\omega)$  tend to be large at high frequencies. This means that the minimisation of the converged error and the error variance are usually not conflicting aims, and  $Q(q)$  is taken as a low-pass filter with a sensibly chosen cut-off frequency.

## 5 SIMULATION AND EXPERIMENTAL RESULTS

### 5.1 Simulation Results

A simulation was carried out to demonstrate the theoretical results. The real, continuous-time system,  $G(s)$ , and its identified model,  $\hat{G}(s)$ , were taken as:

$$G(s) = \frac{20}{(s+1)(s+20)} \quad \hat{G}(s) = \frac{1}{(s+1)}. \quad (54)$$

$G(s)$  and  $\hat{G}(s)$  were then discretized using  $h = 0.1s$  and a zero-order hold to give the discrete-time systems  $G(q)$  and  $\hat{G}(q)$  respectively.  $y_d(t)$  is defined by:

$$y_d(t) = \begin{cases} 1 - \cos(0.1\pi t) & 0 \leq t \leq 20s \\ 0 & 20.1 \leq t \leq 30s \end{cases} \quad (55)$$

Using this  $y_d(t)$  and the specified sampling period gave  $N = 301$ . The load disturbance,  $d_k(t)$ , was taken as a normally distributed, random sequence with  $E\{d_k(t)\} = 0$  and  $\sigma_d^2 = 0.0025$ . The measurement disturbance,  $n_k(t)$ , was taken as a zero-mean, normally distributed, random sequence with variance equal to 0.0025 filtered with a 5th order Butterworth high-pass filter with a cut-off frequency of 2Hz, to simulate high frequency measurement noise. It had a measured variance of 0.0015. The different algorithms analysed in the previous section were tested. 10 iterations were carried out for each algorithm and in order to obtain an estimate of the expected value and variance of the error at a specific time each simulation was repeated 200 times. The expected value and variance at  $t = 15s$ , and the 2-norm value were then calculated for the error at  $k = 10$  for the 200 simulations. The expectation operator was implemented as a mean over the simulations i.e.

$$E\{v_k(t)\} = \frac{1}{N_e} \sum_{j=1}^{N_e} v_k^j(t) \quad (56)$$

where  $N_e$  is the total number of simulations and  $v_k^j(t)$  represents an arbitrary signal at time  $t$ , iteration  $k$  and simulation  $j$ . The iteration number was chosen to allow the algorithm to have converged to a point where the errors due to the disturbances were dominant over the deterministic errors. The time  $t = 15s$  was chosen arbitrarily. Although the disturbances affecting the system were different for the 200 simulations, the same disturbance signals were used for each of the different algorithms. This meant a direct comparison could be made of how each algorithm performed in the presence of the same disturbances.

Table 1 shows the simulation results for the different algorithms. Certain clarifications are perhaps necessary. The algorithm with the forgetting factor was tested using different values for the forgetting factor. Additionally, the different ways of implementing the forgetting factor were tested, firstly when the factor only affects the previous input i.e.  $L = \frac{L'}{1-\alpha}$ , and secondly when it affects the entire algorithm i.e.  $L = L'$ . The filtered version of the algorithm used a 5th order Butterworth low-pass filter with a cut-off frequency of 0.3Hz for  $Q(q)$ . The filter was implemented in a non-causal fashion so as to give zero phase change. The cut-off frequency was chosen to be above that of the highest frequency component of  $y_d(t)$ , which is at 0.05Hz.

From the table the compromise between minimising the variance of the error whilst keeping its expected value small is clearly seen. The standard algorithm has the second smallest expected error value but the second largest error variance. The algorithms using the forgetting factor show the tendencies anticipated from the theoretical results. In the case of the decreasing gain, we see that the variance is much smaller than the standard case, however, the expected value of the error is not so small. This is because, as previously mentioned, the learning rate reduces with each iteration so the algorithm has not converged as quickly as the standard algorithm. The filtered algorithm has the lowest expected error and joint lowest variance, it also has the smallest 2-norm value. This good performance is because the  $y_d(t)$  is limited to low frequencies so allows a low-pass filter to effectively filter out the disturbances at higher frequencies.

Table 1: Expected value and variance of  $\epsilon_{10}(15)$  and  $\|\epsilon_{10}(t)\|_2$  found over 200 simulations

Algorithm	$ E\{\epsilon_{10}(15)\} $	$E\{\tilde{\epsilon}_{10}^2(15)\}$	$\ \epsilon_{10}\ _2$
<b>Standard</b>	0.0065	0.0050	0.0682
<b>Forgetting Factor</b> ( $\alpha = 0.1$ )			
$L = \frac{L'}{1-\alpha}$	0.0987	0.0048	0.1124
$L = L'$	0.1081	0.0044	0.1183
<b>Forgetting Factor</b> ( $\alpha = 0.5$ )			
$L = \frac{L'}{1-\alpha}$	0.3474	0.0048	0.3395
$L = L'$	0.5177	0.0032	0.5018
<b>Forgetting Factor</b> ( $\alpha = 0.9$ )			
$L = \frac{L'}{1-\alpha}$	0.8184	0.0071	0.6461
$L = L'$	0.9317	0.0027	0.8996
<b>Decreasing Gain</b>	0.0266	0.0030	0.0556
<b>Low-pass Filter</b>	0.0045	0.0027	0.0513

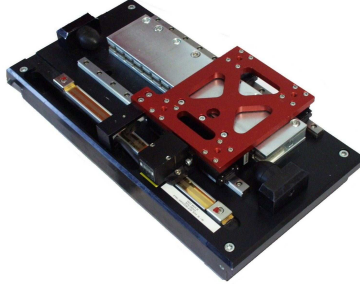


Figure 2: Linear motor (courtesy of ETEL)

## 5.2 Experimental Results

The different ILC algorithms were applied to the tracking control of a linear, permanent magnet, synchronous motor (LPMSM), see Fig. 2. LPMSMs are very stiff and have no mechanical transmission components. They, therefore, do not suffer from backlash and so allow very high positioning accuracy to be achieved. The position of the motor used in the experiment is controlled by a two-degree-of-freedom controller operating at a sampling frequency of 2kHz and tuned to provide robust stability. The motor position is measured by an analog position encoder with a period of  $2\mu\text{m}$ , which is interpolated to obtain a resolution of 0.24nm.

The input,  $u_k(t)$ , computed by the ILC algorithms, is used as the reference signal of the closed-loop system. This means the transfer function  $G(q)$  represents the closed-loop motor system. The desired output position,  $y_d(t)$ , was a series of three low-pass filtered steps, each of amplitude 25mm in the positive direction, followed by a similar series of filtered steps in the negative direction, as can be seen in Figure 3 and has  $N = 8192$ .

An approximate Box-Jenkins model of the closed-loop motor system was identified using standard identification techniques and a PRBS as an input signal. The PRBS was created using a shift register of 10 bits and the resulting signal was repeated eight times

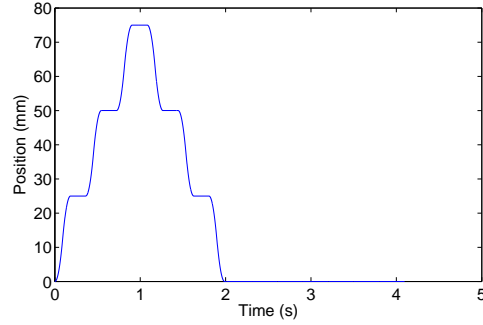


Figure 3: Desired output position,  $y_d$

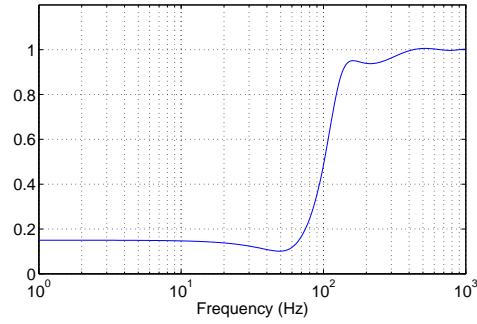


Figure 4: Monotonic convergence condition (8) with  $Q(q) = 1$  and  $\hat{G}^{-1}(q) = 0.85q^6$

giving a total length of 8184 points. The identified model is:

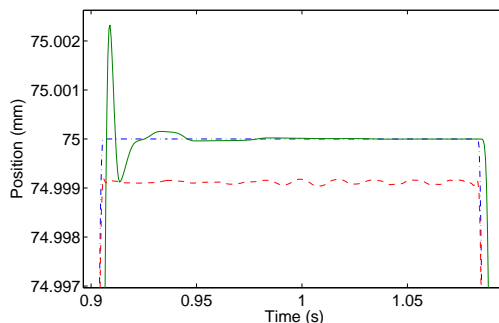
$$G(q) = \frac{0.002008q^3 + 0.0009185q^2 + 0.01972q - 0.009375}{q^4 - 3.104q^3 + 3.739q^2 - 2.058q + 0.4364}. \quad (57)$$

It was used to calculate a phase lead compensator as an approximation to  $\hat{G}^{-1}(q)$ . This gave  $\hat{G}^{-1}(q) = 0.85q^6$ , which satisfied condition (8) up to a frequency of 424Hz, with  $Q(q) = 1$ , see Fig. 4. To assure monotonic convergence it was thus necessary to take  $Q(q)$  as a low-pass filter with a cut-off frequency below 424Hz, 400Hz was chosen. The filter was implemented in a non-causal way so as to give zero-phase shift. A fifth order Butterworth filter was used. Despite the phase lead compensator not being an accurate inverse of the system model it allows reasonably rapid convergence to be attained without going through the laborious process of very accurate modelling. Because a low-pass filter was necessary for deterministic convergence the standard algorithm was not implementable experimentally. The other algorithms, however, were implemented, all being filtered. For each experiment 100 iterations were carried out and each experiment was repeated four times. It is, obviously, not possible to measure  $\varepsilon_k(t)$  in real experiments so the measured error  $e_k(k)$  has to be used for comparisons. For all experiments  $u_0(t) = y_d(t)$  was taken.

For the forgetting factor algorithms a forgetting factor of  $\alpha = 10^{-6}$  was used. This value was chosen in order to allow a reasonable value of the 2-norm of the converged

Table 2:  $\|e_{100}\|_2$  found over 4 experiments

Algorithm	$\ e_{100}\ _2$
<b>Low-pass Filter</b>	$7.4257 \times 10^{-5}$
<b>Forgetting Factor</b> ( $\alpha = 10^{-6}$ )	$3.8603 \times 10^{-4}$
<b>Decreasing Gain</b>	$1.0673 \times 10^{-4}$

Figure 5: Mean experimental results at  $k = 100$  for the forgetting factor algorithm (dashed), before ILC is applied (solid) and  $y_d(t)$  (dot-dash)

measured error,  $\|e_\infty\|_2$ , to be obtained. Expressions (31), (39) and (42) were used in the noiseless, ideal case i.e.  $e_k(t) = \varepsilon_k(t)$  and  $\Delta(q) = 1$  to find upper bounds on  $\|e_\infty\|_2$ . The desired value of  $\|e_\infty\|_2$  was taken as that achieved with the low-pass filtered algorithm. The forgetting factor was of such a small value that  $L' \approx \frac{L'}{1-\alpha}$ , therefore the two variations became, essentially, the same.

Table 2 shows the value of  $\|e_{100}\|_2$  obtained with the different algorithms. It is seen that the decreasing gain algorithm is about 1.4 times greater than that achieved with the filtered version and the forgetting factor algorithm gives a value approximately five times larger.

Fig. 5 shows the mean tracking performance at  $k = 100$  for the forgetting factor algorithm for a small section of the trajectory. It is clear that even with the small  $\alpha$  used a constant error still occurs with the forgetting factor algorithm, explaining the much larger value of  $\|e_{100}\|_2$ . Fig. 6 shows a zoom for the other two algorithms. The decreasing gain algorithm has slightly more oscillation in the overshoot region, due to its slower learning of the deterministic errors, this is probably the cause of its larger 2-norm value. However, it does give a less oscillatory performance in the steady-state region. This, perhaps, is because noise exists at frequencies below the filter cut-off frequency so the decreasing gain algorithm helps reduce its detrimental effect. Reducing the filter cut-off frequency would help reduce sensitivity to noise but prevents as small an expected error from being achieved for this  $y_d(t)$ .

These observations are confirmed in Table 3, where the expected value and variance of the error for a certain time in this region are smaller for the decreasing gain algorithm.

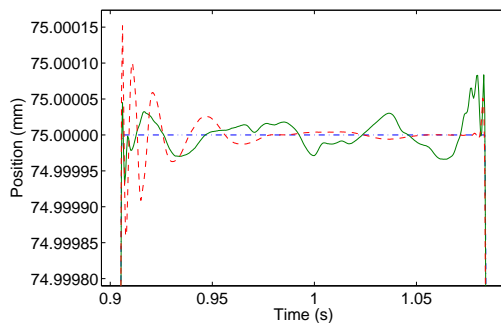


Figure 6: Mean experimental results at  $k = 100$  for the algorithms: decreasing gain (dashed) and low-pass filtered (solid), also  $y_d(t)$  (dot-dash)

Table 3: Expected value and variance of  $e_{100}(1)$  over 4 experiments

Algorithm	$ E\{e_{100}(1)\} (\text{mm})$	$E\{\tilde{e}_{100}^2(1)\}(\text{mm}^2)$
<b>Low-pass Filter</b>	$1.3672 \times 10^{-5}$	$9.1445 \times 10^{-10}$
<b>Forgetting Factor</b> ( $\alpha = 10^{-6}$ )	$8.6169 \times 10^{-4}$	$2.7705 \times 10^{-9}$
<b>Decreasing Gain</b>	$1.4038 \times 10^{-7}$	$6.1343 \times 10^{-12}$

## 6 CONCLUSIONS

New expressions for the expected value and variance of the error in the presence of stochastic disturbances are developed in general, and for a number of specific ILC algorithms. It is found that a trade off between minimising the expected value and variance of the error commonly occurs. When the spectra of the noise and desired output are situated in different frequency regions it is found that a filtered algorithm can give good tracking performance. If the spectra overlap too much, however, an algorithm with a decreasing learning gain is shown to give good robustness to noise and small tracking errors, although it has a slower error convergence rate.

## 7 ACKNOWLEDGMENTS

This work is supported by Office Fédéral de la Formation Professionnelle et de la Technologie under the CTI-projet no. 7049.1 IWS-IW.

The authors acknowledge the collaborations of Ralph Coleman and Michel Mathia from the company ETEL for the real-time experiments.

The authors are very grateful for the helpful comments of the anonymous reviewers.

## References

- [1] S. Arimoto, S. Kawamura, and F. Miyazaki. Bettering operations of robots by learning. *Journal of Robotic Systems*, 1:123–140, 1984.
- [2] K. L. Moore. Iterative learning control: An expository overview. *Applied and Computational Controls, Signal Processing and Circuits*, 1(1):151–214, 1998.



- [3] D.A. Bristow, M. Tharayil, and A.G. Alleyne. A survey of iterative learning control. *IEEE Control Systems Magazine*, 26(3):96–114, June 2006.
- [4] K. L. Moore. Iterative learning control for deterministic systems. In M. Johnson and M.J. Grimble, editors, *Advances in Industrial Control*. Springer-Verlag, Berlin, 1993.
- [5] Z. Bien and J. Xu. *Iterative Learning Control: Analysis, Design, Integration and Application*. Kluwer Academic Publishers, Boston MA, 1998.
- [6] M. Norrlof and S. Gunnarsson. Disturbance aspects of iterative learning control. *Engineering Applications of Artificial Intelligence*, 14(1):87–94, 2001.
- [7] S. Panzieri and G. Ulivi. Disturbance rejection of iterative learning control applied to trajectory tracking for a flexible manipulator. In *European Control Conference*, pages 2374–2379, Rome, Italy, September 1995.
- [8] R. Merry, R. van de Molengraft, and M. Steinbuch. The influence of disturbances in iterative learning control. In *IEEE Conference on Control Applications*, pages 974–979, Toronto, Canada, August 2005.
- [9] G. Heinzinger, D. Fenwick, B. Paden, and F. Miyazaki. Robust learning control. In *28th IEEE Conference on Decision and Control*, pages 436–440, Tampa, Florida USA, December 1989.
- [10] S. Arimoto. Robustness of learning control for robotic manipulators. In *IEEE International Conference on Robotics and Automation*, pages 1528–1533, Cincinnati, Ohio USA, May 1990.
- [11] M.K. Tao, R.L. Kosut, and A. Gurcan. Learning feedforward control. In *IEEE American Control Conference*, pages 2575–2579, Baltimore, Maryland USA, June 1994.
- [12] M. Norrlof and S. Gunnarsson. Disturbance rejection using an ilc algorithm with iteration varying filter. *Asian Journal of Control*, 6(3):432–438, 2004.
- [13] S.S. Saab. A discrete-time stochastic learning control algorithm. *IEEE Transactions on Automatic Control*, 46(6):877–887, June 2001.
- [14] M. Norrlof. An adaptive iterative learning control algorithm with experiments on an industrial robot. *IEEE Transactions on Robotics and Automation*, 18(2):245–251, April 2002.
- [15] S. Gunnarsson and M. Norrlof. On the disturbance properties of high order iterative learning control algorithms. *Automatica*, 42(11):2031–2034, 2006.
- [16] M. Norrlof and S. Gunnarsson. Time and frequency domain convergence properties in iterative learning control. *International Journal of Control*, 75(14):1114–1126, 2002.
- [17] K. L. Moore. Multi-loop control approach to designing iterative learning controllers. In *37th IEEE Conference on Decision and Control*, pages 666–671, Tampa, Florida USA, December 1998.

- [18] S.S. Saab. Optimal selection of the forgetting matrix into an iterative learning control algorithm. In *IEEE International Symposium on Intelligent Control*, pages 1231–1234, Limassol, Cyprus, June 2005.