

## A GENERAL FRAMEWORK FOR ITERATIVE LEARNING CONTROL

Ola Markusson\*, Håkan Hjalmarsson\* and Mikael Norrlöf\* \*

\* Dept. of Signals, Sensors, & Systems,      \*\* Dept. of Electrical Engineering  
 Royal Institute of Technology (KTH)      Linköping University  
 Stockholm, SWEDEN      Linköping, SWEDEN  
 ola.markusson@s3.kth.se    hakan.hjalmarsson@s3.kth.se,    mino@isy.liu.se

**Abstract:** In this contribution we place ILC in the realm of numerical optimization. This clarifies the role played by the design variables and how they affect e.g. convergence properties. We give a model based interpretation of these design variables and also a sufficient condition for convergence of ILC which is similar in spirit to the sufficient and necessary condition previously derived for linear systems. This condition shows that the desired performance has to be traded against modeling accuracy. Finally, one of the main benefits of ILC when non-minimum phase systems are concerned, the possibility of non-causal control, is given a comprehensive coverage.

**Keywords:** Iterative methods, Learning control, Nonlinear systems, Convergence analysis, Non-minimum phase systems

### 1. INTRODUCTION

Iterative Learning Control (ILC) has proven to be a useful tool for feedforward based tracking problems where high precision is desired and where the system is so complex that normal feedback control is insufficient, c.f. robotics. The method can be applied when the reference signal is given over a finite time interval and thereafter repeated. The idea is to use information about the control error from previous iterations to improve performance in the next iteration (Moore, 1999). Many publications have been written on ILC and (Moore, 1993), (Moore, 1999) and (Norrlöf, 2000) provide good introductions. A commonly used example is a robot arm, see e.g., (Norrlöf, 2000), where the trajectory of movement is repeated over and over again.

The system is given by

$$y_k(t) = P(u_k(t)) \quad (1)$$

where  $u_k(t)$  is the input to the system  $P$  and  $y_k(t)$  is the corresponding output, both at the  $k$ th iteration. Here,  $P$  is a general discrete-time single-input single-output (SISO) system which may be linear or a nonlinear<sup>1</sup>. The signal  $y_d(t)$  is the desired output and is given over a finite interval  $t \in [1, N]$ . Provided it exists, we will denote the corresponding input by  $u_d(t)$ , i.e.  $y_d(t) = P(u_d(t))$ .

The system input at the next iteration,  $u_{k+1}(t)$ , is computed as a function of the desired output  $y_d(t)$  and the system output  $y_k(t)$  at the  $k$ th iteration. The main objective is to iteratively refine the input  $u_k(t)$  such that  $y_k(t)$  becomes close to  $y_d(t)$ .

*Remark 1:* Note that if, at convergence,  $u_d(t), \forall t \in [1, N]$  and if the closed loop system is one-to-one then  $u_d(t)$  is the inverse of the system given  $y_d(t)$ .

*Remark 2:* The system is usually already under some kind of feedback control. In this case  $P$  denotes the closed loop system.

### 2. GENERALIZATION AND NOTATION

A first order ILC algorithm can be described by

$$\begin{aligned} u_{k+1}(t) &= Q u_k(t) + L e_k(t) \\ e_k(t) &= y_d(t) - y_k(t) \end{aligned} \quad (2)$$

where  $Q$  and  $L$  denotes linear or nonlinear operators which can be seen as design variables chosen by the user. Notation such as  $u_k$  will be shorthand for  $\{u_k\}_{t=1}^{\infty}$ .

With  $u_i^N = [u_i(1), \dots, u_i(N)]^T \in R^{1 \times N}$  and  $y_k^N$  defined similarly, the algorithm in (2) can also be described in matrix notation as

$$u_{k+1}^N = Q_N (u_k^N + L_N e_k^N) \quad (3)$$

where  $Q_N, L_N \in R^{N \times N}$ . For example, with  $Q = Lq$  ( $q$  denotes the time-shift operator) being causal and linear time-invariant, then  $Q_N$  is given by a lower Toeplitz matrix based on the impulse coefficients  $q_j$ .

<sup>1</sup> With some abuse of notation we will use  $P(u(t))$  to denote the output of a nonlinear dynamical system with  $u(t)$  as input.

### 3. ILC FROM AN OPTIMIZATION PERSPECTIVE

In this section we will take a fresh look at the problem of finding a good feedforward control sequence.

#### 3.1 Numerical optimization

One way to determine a suitable input would be to form some cost function based on the difference between  $y_d$  and  $y$  such as

$$V(u^N) = \frac{1}{2} \|y_d^N - y^N\|^2$$

and try to minimize this function using gradient based numerical optimization methods. This leads to algorithms of the type

$$\begin{aligned} u_{k+1}^N &= u_k^N - \gamma_k R_k \frac{dV(u^N)}{du^N} \\ &= u_k^N + \gamma_k R_k \frac{dy^N}{du^N} (y_d^N - y^N) \end{aligned}$$

where the step-size  $\gamma_k$  is a scalar and  $R_k$  is some matrix. It is well known that the choice of step-size and the matrix  $R_k$  has a profound impact on the properties of the above algorithm.

Now assume that the system is linear such that  $y^N = P_N u^N$  where  $P_N$  is a lower Toeplitz matrix formed from the  $N$  first impulse response coefficients of the system. Then  $dy^N/du^N = P_N$  and hence

$$u_{k+1}^N = u_k^N + \gamma_k R_k P_N (y_d^N - y^N)$$

Here we see clearly that since knowledge of  $P_N$  is required, the algorithm cannot be used directly unless the system is known. Below we will discuss two approaches to circumvent this problem.

#### 3.2 Iterative Learning Control

With  $\gamma_k = 1$  and  $R_k = L_N P_N^{-1}$  we recover the ILC algorithm (2) with  $Q_N$  the identity. Hence we can interpret ILC in the framework above with a *very* special choice of matrix  $R_k$ .

A common choice of  $R_k$  is to use the inverse of the Hessian of  $V$  which in the linear case is  $(P_N P_N^T)^{-1}$ . This gives

$$u_{k+1}^N = u_k^N + P_N^{-1} (y_d^N - y^N)$$

which converges in one step since the cost function is quadratic in  $u^N$  when the system is linear. In ILC this corresponds to using  $L_N = P_N^{-1}$  and  $Q_N$  the identity.

When the system is nonlinear, the cost function is no longer quadratic in  $u^N$  and even if the Hessian is used, convergence in a finite number of steps cannot be expected.

An interesting observation here is that if the ILC algorithm is modified slightly, then convergence in one step can also be obtained. Instead of using the control error  $e_k(t)$  explicitly as in (3) the ILC algorithm can be written as

$$u_{k+1}^N = Q_N (u_k^N + L_N (y_d^N - y_k^N)) \quad (4)$$

For a linear  $L$  this makes no difference compared to (3) but for a nonlinear  $L$  this is not true. Taking  $Q_N$  identity and  $L_N (y_k^N) = P_N^{-1} (y_k^N) \triangleq u_k^N$  gives convergence in one step.

#### 3.3 Design variables in ILC

From the previous discussion in this section, it should be clear that, in terms of convergence rate,  $L_N$  should be taken as an inverse model of the system  $P$ . In the next section we will see that  $L_N$  can be given this interpretation also from a stability point of view. Without loss of generality we will set  $L_N = \hat{P}_N^{-1}$  so that

$$u_{k+1}^N = Q_N (u_k^N + \hat{P}_N^{-1} (y_d^N) - \hat{P}_N^{-1} (y_k^N)) \quad (5)$$

Thus  $Q$  and  $\hat{P}$  are the two design variables and in the following we will examine how they influence stability and convergence of the algorithm. Due to lack of space we will not discuss the inclusion of a step-size parameter in the algorithm.

### 4. CONVERGENCE POINTS

We will begin our analysis of ILC by looking at the possible convergence points of (5). We will use  $u_\infty$  etc to denote the convergence points of the signals. Using operator notation and assuming that  $Q = 1$  gives that  $\hat{P}^{-1}(y_d) = \hat{P}^{-1}(y_\infty)$  and hence, if  $\hat{P}^{-1}$  is bijective, that  $y_\infty = y_d$ , i.e. perfect tracking. In this case, of course, the problem must be well posed in the sense that there must be an input trajectory such that the output matches the desired trajectory  $y_d$ .

If  $Q \neq 1$  but still linear, then

$$\begin{aligned} u_\infty(t) &= Q(q)(u_\infty(t) + \hat{P}^{-1}(y_d(t)) - \hat{P}^{-1}(y_\infty(t))) \\ &= \frac{Q(q)}{1 - Q(q)} (\hat{P}^{-1}(y_d(t)) - \hat{P}^{-1}(y_\infty(t))) \end{aligned} \quad (6)$$

This corresponds to the feedback system in Fig. 1. If in addition  $\hat{P}^{-1}$  and  $P$  are linear we get that the error  $e_\infty(t) = y_d(t) - y_\infty(t)$  can be written as

$$e_\infty(t) = \frac{1}{1 + \frac{Q(q)}{1-Q(q)} \hat{P}^{-1}(q) P(q)} y_d(t). \quad (7)$$

From this we see that both  $Q$  and  $\hat{P}^{-1}$  influence the tracking accuracy. Notice that even if  $\hat{P} = P$  the tracking error will be non-zero when  $Q \neq 1$ .

In general a small tracking error at a particular frequency can be ensured by letting  $Q$  be close to 1 at that frequency. From now on we will assume that  $Q$  is linear.

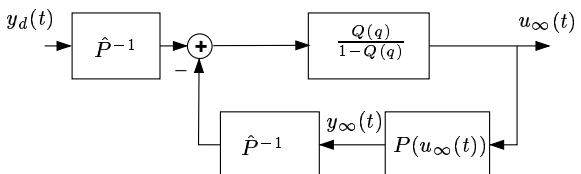


Fig. 1. Block diagram of the ILC algorithm as  $k \rightarrow \infty$  in the case  $Q \neq 1$ .

### 5. CONVERGENCE RESULTS FOR LINEAR SYSTEMS

The ILC convergence criteria for linear systems are given in (Norrlöf, 2000) and can, with our notation  $\hat{P} = L^{-1}$ , be written

$$\left\| Q (1 - \hat{P}^{-1}P) \right\|_{\infty} = \left\| Q \frac{\hat{P} - P}{\hat{P}} \right\|_{\infty} < 1 \quad (8)$$

where  $\|H(e^{i\omega})\|_{\infty} = \sup_{\omega \in [0, \pi]} |H(e^{i\omega})|$ .

This reinforces our interpretation of  $L = \hat{P}^{-1}$  as the inverse of a model  $\hat{P}$  of the system  $P$ . We see that ILC will converge if the relative model error is sufficiently small compared to  $Q$ . In light of Section 4 we see that the tracking accuracy will be limited by the relative model uncertainty. If the relative error is larger than one at some frequency, the tracking cannot be perfect at that frequency.

Notice also that if the system has a zero on the unit circle such that  $P(e^{j\omega}) = 0$  at some frequency, then the stability condition is not satisfied and convergence to zero error cannot be achieved.

## 6. CONVERGENCE RESULTS FOR NON-LINEAR SYSTEMS

The stability and convergence of the ILC method is a central issue and therefore covered in most publications on ILC. Robustness and convergence rates are, e.g., analyzed in (Wang, 1998) and (Saab, 1999) where the convergence criterion is based on certain assumptions. The most limiting assumptions on the system are global Lipschitz conditions. These assumptions do, e.g., not hold for certain robotic applications as pointed out in (Saab, 1999). It is therefore of interest to derive convergence criteria without any global Lipschitz assumptions.

The contraction mapping approach has previously been used to establish convergence properties for ILC using the specific choices of  $Q = 1$  and  $L = 1$ , (Wang and Horowitz, 1985). Below we will first use a feedback interpretation to establish the convergence criterion. This will be followed by a comparison showing that these results can be interpreted as an extension of the results in (Wang and Horowitz, 1985). Thus the similarity between the feedback interpretation and the contraction mapping approach will be highlighted (in Section 6.2). A motivation for using the feedback approach is that it provides a stepwise derivation of the criterion which gives valuable insight. The approach taken will also provide connections to the results in the linear domain.

### Boundedness

Consider first the block diagram of the ILC algorithm (5) depicted in Fig. 2.

To establish a condition for boundedness of the sequence  $\{u_k\}_{k=1}^{\infty}$  we first let  $\|\cdot\|_2$  denote the two-norm defined by  $\|u\|_2 = \sqrt{\sum_{t=0}^{\infty} |u(t)|^2}$ . Let us further define the concept of BIBO stability in the  $\ell_2$  sense ( $\ell_2$ -stable).

#### Definition 1. BIBO $\ell_2$ -stability

A system is BIBO  $\ell_2$ -stable if the output has a finite two-norm provided that the input has a finite two-norm, (Vidyasagar, 1993).

*Theorem 2.* Assume that the system  $P$  is BIBO  $\ell_2$ -stable. Then the system in (5) and Figure 2 is BIBO  $\ell_2$ -stable (in the iteration domain  $k$ ) if

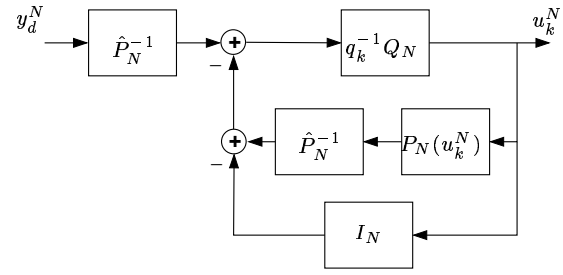


Fig. 2. Block diagram of the ILC algorithm in the nonlinear case given by (5). The system is denoted by  $P_N$  transforming the vector  $u_k^N$  into  $y_k^N$ , the model inverse  $\hat{P}_N^{-1}$  acts in similar way. Note that the shift operator  $q_k$  acts w.r.t. the iteration index  $k$  ( $q_k^{-1}u_k(t) = u_{k-1}(t)$ ).

$$\alpha\beta < 1,$$

where

$$\alpha = \sup_{u \neq 0} \frac{\|Qu\|_2}{\|u\|_2} = \|Q\|_{\infty}, \quad (9)$$

where  $Q$  is a linear mapping, and

$$\beta = \sup_{u \neq 0} \frac{\|u - \hat{P}^{-1}(P(u))\|_2}{\|u\|_2}. \quad (10)$$

*Proof:* See (Markusson, 2002).  $\square$

### Existence

We next ask the question when  $u_{\infty}(t)$  and the corresponding output  $y_{\infty}(t) = P(u_{\infty}(t))$  defined by (6), c.f. Figure 1, are well-defined in the case  $Q \neq 1$ . The following theorem applies in this case.

*Theorem 3.* Assume that the conditions of Theorem 2 hold, that  $Q \neq 1$  and that  $\hat{P}^{-1}$  is BIBO stable. Then the system (6) (c.f. Figure 1) is BIBO  $\ell_2$ -stable and the corresponding signals  $u_{\infty}$  and  $y_{\infty}$  are well defined.

*Proof:* See (Markusson, 2002).  $\square$

The case when  $Q = 1$  will also be covered below.

### 6.1 Convergence

In this section we present a convergence result very similar to (8) in the linear case. Suppose that  $y_d$  is bounded, that the conditions of Theorem 2 are satisfied such that the system (5) is BIBO stable and suppose also that  $u_{\infty}$  is well defined, meaning that the conditions for existence are fulfilled as given above.

Under these assumptions it is possible to define  $\Delta_k(t) \triangleq u_k(t) - u_{\infty}(t)$  and

$$\begin{aligned} \Phi_N(u_{\infty}^N, \Delta_k^N) &\triangleq \\ &\hat{P}^{-1} (P_N(\Delta_k^N + u_{\infty}^N)) - \hat{P}^{-1} (P_N(u_{\infty}^N)). \end{aligned} \quad (11)$$

Now, combining (4) and (11), gives

$$\begin{aligned} \Delta_{k+1}^N &= u_{k+1}^N - u_{\infty}^N \\ &= Q_N \Delta_k^N - Q_N \Phi_N(u_{\infty}^N, \Delta_k^N). \end{aligned} \quad (12)$$

The system in (12) is an unforced system, initialized with a nonzero initial condition ( $\Delta_0^N \neq 0$ ). Convergence of the ILC algorithm in the nonlinear case will be achieved if the origin is a globally asymptotically stable equilibrium of the system in (12).

*Theorem 4.* Assume that the conditions in Theorem 2 are satisfied such that (4) is BIBO stable and  $u_\infty$  is well-defined for  $Q \neq 1$ . For  $Q = 1$  assume that the system is invertible, for the desired output  $y_d$ , such that  $u_\infty = u_d$  exists.

Assume that

$$\alpha\gamma < 1 \quad (13)$$

where  $\alpha$  is the induced norm of  $Q$  (9) and  $\gamma$  is the induced norm of  $1 - \Phi(\cdot)$  given by

$$\gamma = \sup_{\Delta \neq 0} \frac{\|\Delta - \Phi(u_\infty, \Delta)\|_2}{\|\Delta\|_2}. \quad (14)$$

The input error  $u_\infty^N - u_{k+1}^N$  in the ILC algorithm in (5) fulfills  $\|u_\infty^N - u_{k+1}^N\| < \alpha\gamma\|u_\infty^N - u_k^N\|$  i.e., it is a monotonically decreasing difference with

$$\lim_{k \rightarrow \infty} \sup \|u_\infty^N - u_k^N\| = 0. \quad (15)$$

*Proof:* See (Markusson, 2002).  $\square$

*Remarks:*

- When  $\hat{P}^{-1}$  and  $P$  are linear, the norms (10) and (14) are identical and the boundedness, existence and convergence conditions all become

$$\|Q\|_\infty \|1 - \hat{P}^{-1}P\|_\infty < 1.$$

Thus the sufficient condition here is very close in spirit to the necessary and sufficient condition (8).

- Notice the similarity between (14) and the relative model error

$$\|1 - \hat{P}^{-1}P\|_\infty,$$

in the linear case. Hence, also in the nonlinear case  $\hat{P}$  should approximate the system dynamics as well as possible in a relative sense. Notice, however that in the perfect tracking case ( $Q = 1$ ) a relative model error of up to 100% can be tolerated. This is, probably, one of the main reasons for the success of ILC. An accurate model is not required!

- The guaranteed convergence rate is geometric with rate  $\alpha\gamma$ . Hence, for a given modeling accuracy  $\gamma$ , the convergence rate can be increased by reducing  $\alpha$ , i.e. by increasing the tracking error.

## 6.2 The contraction mapping approach

There is a clear relationship between the results above and the contraction mapping theorem (Vidyasagar, 1993). Recall the ILC algorithm represented as

$$u_{k+1}^N = Q_N \left( u_k^N - \hat{P}_N^{-1}(P_N(u_k^N)) \right) + Q_N \hat{P}_N^{-1}(y_d^N),$$

which is rewritten as

$$\begin{aligned} u_{k+1}^N &= Q_N \left( u_k^N - \hat{P}_N^{-1}(P_N(u_k^N)) \right) + Q_N \hat{P}_N^{-1}(y_d^N) \\ &\triangleq \tilde{T}(u_k^N) + Q_N \hat{P}_N^{-1}(y_d^N) \\ &\triangleq T(u_k^N). \end{aligned} \quad (16)$$

At convergence, (16) satisfies  $u_\infty^N = T(u_\infty^N)$ , hence,  $u_\infty^N$  is a fixed point to the mapping  $T$ . The question is now under what conditions this is true and under what conditions the fixed point can be reached using the iterations  $u_{k+1}^N = T(u_k^N)$ . The contraction mapping theorem (Vidyasagar, 1993) states that assuming that there exists a fixed positive constant  $\rho < 1$  such that

$$\|T(x) - T(y)\| \leq \rho \|x - y\|, \quad \forall x, y \in S, \quad (17)$$

where  $S$  denotes the set such that  $x, T(x) \in S$ . Under these conditions there exists only one fixed point  $x^*$ , satisfying  $x^* = T(x^*)$ , and this fixed point will be reached at convergence by  $x_{k+1} = T(x_k)$ . A strength of the contraction mapping approach is that it implicitly gives conditions for both existence and convergence.

To compare the criterion for convergence derived by the feedback approach we first consider the criterion for a bounded solution. In Theorem 2 it was stated that the mapping

$$u_{k+1}^N = Q_N \left( u_k^N - \hat{P}^{-1}(P(u_k^N)) \right) + Q_N \hat{P}^{-1}(y_d^N)$$

is bounded if  $\alpha\beta < 1$ , where  $\alpha$  is defined in (9) and  $\beta$  in (10). This is similar to the condition

$$\|\tilde{T}(u_k^N)\| \leq \eta \|u_k^N\|, \quad 0 < \eta < 1,$$

which is a relaxation of (17) with  $T$  as in (16),  $x = u_k^N$  and with  $y = 0$ . In this case

$$\|u_{k+1}^N\| \leq \eta \|u_k^N\| + \|Q_N \hat{P}_N^{-1}(y_d^N)\|,$$

assuming that  $\|Q_N \hat{P}_N^{-1}(y_d^N)\| < \infty$ .

The criterion for existence of a fixed point in the case  $Q \neq 1$  is stated in Theorem 3 but there does not seem to be any specific relation to the contraction mapping approach in this case.

Assuming that a fixed point  $u_\infty^N$  exists, the condition of the contraction mapping approach can be relaxed to

$$\|T(u_k^N) - T(u_\infty^N)\| \leq \eta \|u_k^N - u_\infty^N\|, \quad 0 < \eta < 1,$$

which is similar to the criterion in Theorem 4.

By strengthening the condition in Theorem 4 to (17), i.e.,

$$\begin{aligned} \left\| Q_N(u_k^N - z_k^N) - Q_N \hat{P}_N^{-1}(P_N(u_k^N)) - \right. \\ \left. Q_N \hat{P}_N^{-1}(P_N(z_k^N)) \right\| \leq \eta \|u_k^N - z_k^N\|, \\ 0 < \eta < 1, \end{aligned} \quad (18)$$

for any inputs  $u_k^N$  and  $z_k^N$ , the assumption that  $u_\infty^N = u_d^N$  exists when  $Q = 1$  can be dispensed with. The existence of the inverse then follows from the contraction mapping theorem, i.e. (18) implies existence of the inverse.

It can further be pointed out that if

$$\sup_x \left\| \frac{dT(x)}{dx} \right\| \leq \alpha < 1,$$

on a convex set  $S$  then  $T$  is a contraction mapping, which follows from the mean value theorem (Luenberger, 1969). With  $T$  as in (16), we have

$$\frac{dT}{du^N} = \frac{d\hat{T}}{du^N} = Q_N \left( I_N - \frac{d}{du^N} \hat{P}_N^{-1}(P_N(u^N)) \right).$$

Now define the sequence  $\delta(t) = u(t) - \hat{P}^{-1}(P(u(t)))$ , then  $\frac{d\delta^N}{du^N} = I_N - \frac{d}{du^N} \hat{P}_N^{-1}(P_N(u^N))$ . Since

$$\left\| \frac{d\delta^N}{du^N} \right\| \leq \lim_{N \rightarrow \infty} \left\| \frac{d\delta^N}{du^N} \right\|,$$

we now have the following result.

*Corollary 6.1.* Assume that  $\hat{P}^{-1}$  and  $P$  are two times continuously differentiable. Define

$$\bar{\gamma} = \lim_{N \rightarrow \infty} \sup_{u^N} \left\| \frac{d\delta^N}{du^N} \right\|$$

and let  $\alpha$  be the induced norm of  $Q$  (9). Then the conclusions of Theorem 4 hold if

$$\alpha \bar{\gamma} < 1.$$

*Proof:* Follows from the discussion preceding the corollary.

It can further be noted that the condition in Theorem 4 and Corollary 6.1 is a generalization of a condition in (Wang and Horowitz, 1985). There ILC algorithm

$$u_{k+1}(t) = u_k(t) + y_d(t) - y_k(t), \quad (19)$$

is analyzed, hence,  $Q = L = 1$ . Convergence of (19) is guaranteed if

$$\sup_{u^N} \left\| I_N - \frac{dP_N(u^N)}{du^N} \right\| < 1, \quad (20)$$

for all inputs  $u \in S$  where  $S$  is a convex subset of  $C[1, N]$ . Here,  $C[1, N]$  denotes a normed linear space consisting of continuous functions on the interval  $[1, N]$ .

## 7. NON-CAUSAL ILC

In (2) the inverse of the model is used to iteratively find the inverse of the system. If the model is non-minimum phase the inverse will be unstable if causal filtering is used. However, by applying non-causal filtering (exploring the batch nature of ILC) the inverse will be stable. This is one *very* interesting and useful feature of ILC. Let us start by considering the linear case.

### *Inversion of linear systems*

Consider a bounded signal  $u(t)$  with  $z$ -transform  $U(z)$  filtered through a stable, *albeit not necessarily minimum-phase*, linear system  $G(z)$  and denote the resulting output  $y(t)$  with transform  $Y(z)$ . Then,  $Y(z) = G(z)U(z)$  in the region of convergence which under the given assumptions is a disc which includes the unit disc so that  $y(t)$  is bounded. Consequently it holds that  $U(z) =$

$G(z)^{-1} Y(z)$ . Since both  $u$  and  $y$  are bounded,  $G^{-1}$  is here a stable operator, i.e. its region of convergence includes the unit disc. Hence, when  $G(z)$  has zeros outside the unit circle,  $G^{-1}(z)$  must be interpreted as a *stable* but *non-causal* operator and to recover  $u$  from  $y$  non-causal filtering has to be performed. This is done as follows: Factorize  $G(z)$  as

$$G(z) = G_+(z)G_-(z) \quad (21)$$

where  $G_+(z)$  includes the minimum phase zeros and  $G_-(z)$  includes the non-minimum phase zeros. In this case the stable inverse of  $G_+(z)$  will be causal while the stable inverse of  $G_-(z)$  is anti-causal and  $u$  is recovered by first filtering  $y$  through the stable causal filter  $G_+^{-1}(z)$

$$w(t) = G_+^{-1} y(t)$$

and then anti-causal filtering of this result

$$u(t) = G_-^{-1} w(t).$$

The latter can be performed by filtering the reversed sequence  $p(t) = w(N - t)$  ( $N$  is the total number of data) through the stable causal filter  $v(t) = G_-^{-1}(q^{-1})p(t)$  and then reversing the result again, i.e.  $u(t) = v(N - t)$ , see e.g., (Oppenheim and Schaffer, 1989) for more details.

### *Nonlinear systems*

Applying the inverse of nonlinear models to (2) can be complicated, especially when the model is non-minimum phase. We therefore suggest that a linear approximation of the system (or model) is used for the inversion in (2). Hence, in the nonlinear case it is the linearized model which is inverted, by the method described above, and applied to (2).

### *Related work*

In (Sogo *et al.*, 2000) an ILC method for nonlinear non-minimum phase systems is presented based on optimization. The optimization is solved using an adjoint system which is time-reversed, hence, stable in the case of maximum phase systems. (For a linear system maximum phase implies that all zeros are outside the unit circle, in the nonlinear case the poles of the linearized zero-dynamics are outside the unit circle.) The idea of using time-reversed filtering for maximum phase systems is also explored in (Choi and Jeong, 2001).

The idea of using a linear model which is split into a causal and an anti-causal part is also suggested in (Ghosh and Paden, 1999) for continuous time systems. In (Jeong and Choi, 2001) a similar matrix based method is presented. However, the matrix based non-causal filtering introduces limitations for certain systems.

## 8. MODELS SUITED FOR ILC

We have seen that  $\hat{P} = L^{-1}$  plays a crucial role in the properties of ILC and that this quantity can be interpreted as a model of the system which, both from a stability and convergence rate perspective, should be chosen *as close as possible to the true system*.

### *8.1 Using nonlinear model*

First of all, if a nonlinear model is available and its inverse suitable for implementation, that

inverse should, of course, be used. However, if the nonlinear model is, e.g., non-minimum phase, it may be difficult to compute a stable inverse. In this case a linear model can be preferable. One option is then to use a linearization of the nonlinear model at the origin. This linearization can be used if the difference  $e_k(t) = y_d(t) - y_k(t)$  is small. Here, the notion small implies that the higher order terms of a series expansion of  $P^{-1}(e_k(t))$  are negligible. As we have seen in Section 7 linear models, even non-minimum phase ones, are easily inverted. It is further assumed that  $Q$  is chosen such that convergence is maintained.

## 8.2 Identification of linear models for ILC

If, initially, a model is unknown, it is natural to use the data from the ILC-iterations to identify a model. In the case of a nonlinear system the identified linear model will depend on the specific set of input-output data. Hence, the analysis in Section 6 suggests that a linearization of the system around the current trajectory should be used. Related work are found in, e.g., (Nijse *et al.*, 2001) where a subspace method is presented for the identification at successive iterations when applying ILC on linear systems.

## 9. NUMERICAL EXAMPLE

Let us illustrate the tradeoff between model accuracy and the design variable  $Q$  to obtain convergence. In this example we consider an inverted pendulum on a cart and the unstable system is stabilized by state-feedback. Details and further results are provided in (Markusson, 2002).

The position of the cart is considered as the output of the system and the reference trajectory is accordingly the desired position of the cart. No model is considered to be known prior to applying the first ILC iteration. A linear model was therefore identified using data from the first ILC iteration. A model was estimated as

$$\hat{P}(z) = \frac{10^{-5}(-0.038z^2 + 0.977z - 0.99)}{z^4 - 3.88z^3 + 5.65z^2 - 3.66z + 0.89},$$

and applied to the ILC algorithm. This model is non-minimum phase and the factorization suggested in Section 7 is utilized.

From the first numerical experiments it followed that convergence could not be obtained with  $Q = 1$ . It was then assumed that the model is not so good for high frequencies and by letting  $Q$  be a low-pass filter convergence could be obtained. The tracking results are shown in Figure 3. We conclude from the good tracking properties that a linear model, identified at the first iteration, is well suited to obtain a satisfying result. By letting  $Q(z)$  be a low pass filter, model errors at higher frequencies were neglected.

## 10. CONCLUSIONS

In this contribution we have highlighted the role of the design variables in ILC. We have also shown that ILC can be placed in the realm of numerical optimization. We have presented conditions for boundedness and convergence for non-linear systems which are very close in spirit to the necessary and sufficient conditions that previously have been derived for linear systems. The analysis

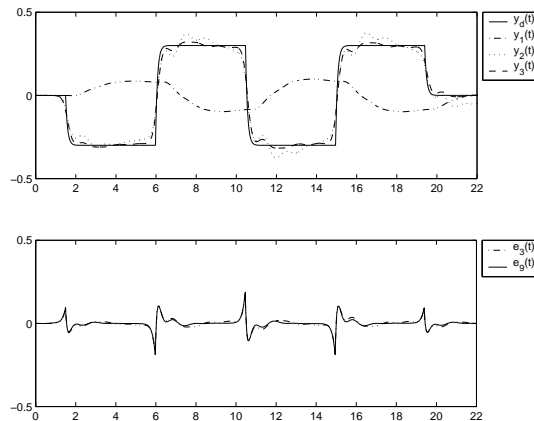


Fig. 3. Top: The desired output and the actual outputs at different iterations for the inverted pendulum. Bottom: Tracking errors.

shows that a relative model error of up to 100% can be tolerated, even when perfect tracking is desired. This is probably one of the reasons for the success of ILC in applications. The benefits of non-causal filtering for non-minimum phase systems has been elaborated upon and the implementational aspects of this covered.

## REFERENCES

- Choi, C.H. and G.M. Jeong (2001). Perfect tracking for maximum phase nonlinear systems by iterative learning control. *International Journal of Systems Science*.
- Ghosh, J. and B. Paden (1999). Iterative learning control for nonlinear non-minimum phase plants with input disturbances. In: *Proceedings of the American Control Conference*. San Diego, California, USA. pp. 2584–2589.
- Jeong, G.M. and C.H. Choi (2001). Iterative learning control with advanced output data for nonminimum phase systems. In: *Proceedings of the American Control Conference (ACC)*. pp. 890–895.
- Luenberger, D.G. (1969). *Optimization by Vector Space Methods*. Wiley.
- Markusson, O. (2002). Model and System Inversion with Applications in Nonlinear System Identification and Control. PhD thesis. Royal Institute of Technology. Stockholm, Sweden, available at [www.s3.kth.se/control](http://www.s3.kth.se/control).
- Moore, K.L. (1993). *Iterative learning control for Deterministic Systems, Advances in Industrial Control*. Springer-Verlag.
- Moore, K.L. (1999). *Applied and Computational Controls, Signal Processing, and Circuits*. Chap. 4. Vol. 1. Editor B.N. Datta, Springer-Verlag.
- Nijse, G., M. Verhagen and N.J. Doelman (2001). A new subspace based approach to iterative learning control. In: *Proceedings of the European Control Conference (ECC)*. pp. 3375–3380.
- Norrlöf, M. (2000). Iterative Learning Control Analysis, Design and Experiments. PhD thesis. Linköping University.
- Oppenheim, A.V. and R.W. Schaffer (1989). *Discrete-Time Signal Processing*. Prentice-Hall.
- Saab, S.S. (1999). Robustness and convergence rate of a discrete-time learning control algorithm for a class of nonlinear systems. *International Journal of Robust and Nonlinear Control* **9**, 559–571.
- Sogo, T., K. Kinoshita and N. Adachi (2000). Iterative learning control using adjoint systems for nonlinear non-minimum phase systems. In: *IEEE Conference on Decision and Control*. pp. 3445–3446.
- Vidyasagar, M. (1993). *Nonlinear Systems Analysis*. Prentice-Hall.
- Wang, D.W. (1998). Convergence and robustness of discrete time nonlinear systems with iterative learning control. *Automatica* **34**(11), 1445–1448.
- Wang, S.H. and I. Horowitz (1985). Create - a new adaptive technique. In: *In Proceedings of 19:th Annual Conference on Information Sciences and Systems*.