On the Design of ILC Algorithms Using Optimization

Svante Gunnarsson, Mikael Norrlöf
Department of Electrical Engineering
Linköping University, S-581 83 Linköping, Sweden
www: http://www.control.isy.liu.se
email: svante@isy.liu.se, mino@isy.liu.se

1999-12-14

Submitted to Automatica as a Brief Paper. Technical reports from the Automatic Control group in Linköping are available by anonymous ftp at the address 130.236.20.24 (ftp.control.isy.liu.se/pub/Reports/). This report is contained in the compressed postscript file 2209.ps.gz.
On the Design of ILC Algorithms Using Optimization *

Svante Gunnarsson † Mikael Norrlöf
Department of Electrical Engineering,
Linköping University,
SE-58183 Linköping, Sweden
svante@isy.liu.se, mino@isy.liu.se
Fax: +46-13-282622

Abstract

Design of Iterative Learning Control (ILC) algorithms using optimization is considered. By forming a quadratic criterion in the control error and the input signal using a nominal model of the system an ILC algorithm is derived. Special attention is paid to the frequency domain properties of the algorithm and to how it is able to handle non-minimum phase systems. A numerical example and an experiment carried out on a real industrial robot are presented.

Keywords: Iterative Learning Control, optimization, robotics

---

*This work was supported by CENIIT at Linköping University and by ABB Robotics within ISIS at Linköping University.
†Corresponding author
1 Introduction

The purpose of this paper is to investigate some properties of an ILC algorithm derived using optimization. The optimization approach to ILC is not new but the aim here is to show some new aspects of this approach. Parts of the results presented here can also be found in (Gunnarsson and Norrlöf 1999).

The general setup is of standard ILC type, i.e. the system to be controlled is carrying out the same operation repeatedly, the desired operation is carried out during a finite time interval and the purpose of ILC is to obtain good servo properties. General introductions to the area of ILC are given in e.g. (Moore 1993) and (Moore 1998) and a collection of recent results is found in (Bien and Xu 1998).

More specifically this paper deals with ILC applied to linear SISO systems working in discrete time. The general system description will then be

\[ y_k(t) = T_r(q)r(t) + T_u(q)u_k(t) + T_d(q)d_k(t) + T_n(q)n_k(t) \]  

(1)

where \( u_k(t) \) and \( y_k(t) \) denote the ILC input signal and the output signal respectively. Furthermore \( r(t), d_k(t) \) and \( n_k(t) \) denote the desired output (reference) signal, a load disturbance and a measurement disturbance respectively. The subscript \( k \) denotes iteration number. The reference signal is the same in all iterations while the other signals will change from iteration to iteration. All signals are defined on a finite time interval \( t = 0, \ldots , N \). Finally \( T_r(q), T_u(q), T_d(q) \) and \( T_n(q) \) are stable discrete time filters.

The formulation in equation (1) is taken from (Norrlöf 1998), and it covers a wide class of situations ranging from an open loop control problem to a closed loop system operating under both feed-back and feed-forward control as depicted in Figure 1. In Figure 1 the signal \( u_k(t) \) represents a signal added to the signals normally generated in the control system. ILC is used as a complement to the conventional robot control system. A slight modification of the system structure shown in Figure 1 is to let the ILC input signal be used as a feed-forward signal to the control signal generated by the feed-back and feed-forward parts of the controller. This just corresponds to a redefinition of the transfer operator \( T_u(q) \) in Equation (1).

The fundamental problem in ILC is to design an update algorithm for the input signal \( u_k(t) \) such that the error \( e_k(t) = r(t) - y_k(t) \) is minimized in some appropriate sense. A general updating equation is given by

\[ u_{k+1}(t) = Q(q)(u_k(t) + L(q)e_k(t)) \]  

(2)

where \( Q(q) \) and \( L(q) \) are linear, possibly non-causal, filters. The choice of \( Q(q) \) and \( L(q) \) are the main issues in the design of an ILC algorithm. The aim in this paper is to show that an ILC algorithm derived using optimization can be seen as a particular choice of the filters \( L(q) \) and \( Q(q) \). The paper is organized as follows. In Section 2 an ILC algorithm is derived using optimization. Then in Section 3 it is shown how this type of ILC algorithm can deal with non-minimum phase system. In Section 4 the frequency domain properties of the algorithm are investigated and these properties are further shown in Section 5 where a numerical example is presented. In Section 6 the ILC algorithm is applied to an industrial robot with good results. Finally some conclusions are given in Section 7.
2 ILC Using Optimization

The optimization approach to ILC is not new, and previous contributions can be found in e.g. (Gorinevsky et al. 1995), (Frueh and Phan 1998), (Lee and Lee 1998), (Amann et al. 1995b) and (Amann et al. 1995a). As mentioned in the introduction the aim of this paper is to investigate some specific aspects of this approach. For completeness however a derivation of the ILC algorithm is presented here.

2.1 Notations

Introduce the vectors

\[
\begin{align*}
Y_k &= (y_k(0), \ldots, y_k(N))^T \\
U_k &= (u_k(0), \ldots, u_k(N))^T \\
R &= (r(0), \ldots, r(N))^T \\
D_k &= (d_k(0), \ldots, d_k(N))^T \\
N_k &= (n_k(0), \ldots, n_k(N))^T \\
E_k &= R - Y_k
\end{align*}
\]

where \( k \) denotes the iteration number and \( t = 0, \ldots, N \) denote the sampling points. Using these notations the system can be described by the equation

\[
Y_h = T_r R + T_u U_k + T_d D_k + T_n N_k
\]
where \( T_n \) is a matrix formed by the impulse response coefficients of the transfer operator \( T_u(q) \), i.e.

\[
T_n = \begin{pmatrix}
  t_0 & 0 & 0 & \ldots & 0 \\
  t_1 & t_0 & 0 & \ldots & 0 \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  t_N & t_{N-1} & \ldots & t_1 & t_0
\end{pmatrix}
\]  

(5)

and \( T_r, T_d \) and \( T_n \) are defined analogously. In equation (4) the matrix elements along the diagonals are constant but this is not necessary. In case the system is time varying in the sense that the dynamics change during one iteration it is straightforward to let the coefficients in \( T_u \) vary along the diagonals. It is however assumed that the same \( T_n \) is valid in each iteration, i.e. slow changes in the system to be controlled are not covered. For completeness the diagonal element \( t_0 \) is included, but the discussion is not restricted to zero relative degree.

The matrices in equation (4) will depend on the structure of the control system. In, for example, open loop control in a disturbance free situation the matrices in Equation (4) are given by

\[
T_r = T_d = T_n = 0 \quad T_u = G
\]  

(6)

where \( G \) is the matrix containing the pulse response coefficients of the system to be controlled. In this case the ILC input signal \( U_k \) is also the actual input to the system.

A more general situation is obtained when the system is controlled using a two-degrees of freedom controller structure and \( U_k \) is used as a feed-forward signal added to the control signal generated by the controller. In such a case the matrices in Equation (4) are given by

\[
T_r = SGF_r \quad T_u = SG \quad T_n = S \quad T_d = SG
\]  

(7)

where

\[
S = (I + GF_y)^{-1}
\]  

(8)

and \( F_r \) and \( F_y \) are the matrices corresponding to the controller transfer functions.

A third structure, which is the one that will be studied here, is the one shown in Figure 1. The matrices in equation (4) are then given by

\[
T_r = SG(F_f + F) \quad T_u = SGF \quad T_n = S \quad T_d = SG
\]  

(9)

where \( F_f \) is the matrix corresponding to the feed-forward transfer operator \( F_f \).

2.2 Criterion minimization

The idea is now to determine \( U_{k+1} \) such that the error \( E_{k+1} \) becomes as small as possible by minimizing the criterion

\[
J = E_{k+1}^{T} W_e E_{k+1} + U_{k+1}^{T} W_u U_{k+1}
\]  

(10)

where \( W_e \) and \( W_u \) are weight matrices determining the trade off between performance and input energy. The weight matrices can be used for both time and frequency weighting but this possibility will not be exploited here. The criterion is minimized subject to the constraint

\[
(U_{k+1} - U_k)^{T} (U_{k+1} - U_k) \leq \delta
\]  

(11)
Introduction of a Lagrange multiplier yields the criterion
\[ \dot{J} = E_{k+1}^T W_e E_{k+1} + U_{k+1}^T W_u U_{k+1} + \lambda ((U_{k+1} - U_k)^T (U_{k+1} - U_k) - \delta) \] (12)
and straightforward differentiation of \( \dot{J} \) then gives
\[ -T_u^T W_e E_{k+1} + W_u U_{k+1} + \lambda (U_{k+1} - U_k) = 0 \] (13)
Equation (4) implies that \( E_{k+1} \) is given by
\[ E_{k+1} = (I - T_u)R - T_u U_{k+1} - T_u D_{k+1} - T_u N_{k+1} \] (14)
Using a statistical description of \( D_k \) and \( N_k \) i.e. the load and measurement disturbances it is possible to use a prediction of the error in the next iteration as done in e.g. (Lee and Lee 1998). This possibility will however not be utilized here and the disturbances will therefore be predicted with there mean values which are assumed to be zero. The error \( E_{k+1} \) will therefore be represented as
\[ E_{k+1} = (I - T_u)R - T_u U_{k+1} \] (15)
where \( T_u \) denotes a nominal model of the closed loop system. This implies that some a priori knowledge of the system to be controlled is available. Using (15) in equation (13) gives
\[ -T_u^T W_e (I - T_u)R + T_u^T W_e T_u U_{k+1} + W_u U_{k+1} + \lambda (U_{k+1} - U_k) = 0 \] (16)
which implies
\[ U_{k+1} = (W_u + \lambda \cdot I + T_u^T W_e T_u)^{-1}(\lambda U_k + T_u^T W_e (I - T_u)R) \] (17)
Using
\[ (I - T_u)R = E_k + T_u U_k \] (18)
the equation can be reformulated into
\[ U_{k+1} = (W_u + \lambda \cdot I + T_u^T W_e T_u)^{-1}(\lambda \cdot I + T_u^T W_e T_u)U_k + T_u^T W_e E_k \] (19)
i.e.
\[ U_{k+1} = Q(U_k + LE_k) \] (20)
where
\[ Q = (W_u + \lambda \cdot I + T_u^T W_e T_u)^{-1}(\lambda \cdot I + T_u^T W_e T_u) \] (21)
and
\[ L = (\lambda \cdot I + T_u^T W_e T_u)^{-1}T_u^T W_e \] (22)
The updating matrices \( Q \) and \( L \) hence depend on the nominal model \( T_u \) and the weight matrices \( W_u \) and \( W_e \). The Lagrange multiplier \( \lambda \) is not computed explicitly but instead used as a design variable.

In equation (20) the error signal is formed using the nominal model of the system, while in real use the actual error signal from the system is used. Introducing
\[ E_{k+1}^0 = R - Y_k^0 \] (23)
where \( Y_k^0 \) is the output from real system leads to
\[ U_{k+1} = Q(U_k + LE_k^0) \] (24)
where \( Q \) and \( L \) are given by (21) and (22).
Remark 2.1 Putting $W_u = 0$, i.e. removing the penalty on the input signal equation (21) implies that $Q = I$, i.e. an update equation without any weighting of the previous input signal. This updating is also obtained by minimizing a criterion of the type

\[
J = E_k^T E_{k+1} + \rho (U_{k+1} - U_k)^T (U_{k+1} - U_k)
\]

as studied in e.g. (Amann et al. 1995b) and (Amann et al. 1995a). As will be seen below the use of $W_u > 0$ i.e. $Q \neq I$ is useful, both for robustness reasons and when dealing with non-minimum phase systems.

Remark 2.2 By instead putting $\lambda = 0$, i.e. removing the constraint on the update step the computation of $U_{k+1}$ can be expressed as

\[
U_{k+1} = (T_u^T W_u T_u + W_u)^{-1} T_u^T W_u R
\]

i.e. the computation of the input vector becomes a one-step procedure.

Remark 2.3 Third, choosing both $W_u = 0$ and $\lambda = 0$ results in the choice (assuming that the inverse exists)

\[
U_{k+1} = T_u^{-1} R
\]

which corresponds to the choice $L(q) = T_u^{-1}(q)$ in equation (2).

2.3 Alternative representations of the input

In the general algorithm, equation (24), derived in the previous section the ILC input is represented by the vector $U$, i.e. the input signal in all sampling points. An interesting reformulation of the ILC algorithm can be done if the input is represented in an alternative form.

Assume therefore that the ILC input is formed as a linear combination of column vectors $w_i$ according to

\[
U = \sum_{i=0}^{m} c_i w_i = WC
\]

where $W$ is a matrix formed using the column vectors $w_i$, i.e.

\[
W = (w_0 \ldots w_m)
\]

and $C$ is a row vector containing the coefficients $c_i$, i.e.

\[
C = (c_0 \ldots c_m)
\]

The standard ILC case corresponds to the situation when $c_i = u(i)$ and $w_i$ are unit pulses, i.e. vectors where all elements are zero except for element $i$, which is one. This corresponds to the case $W = I$. Another choice is to let $w_i$ represent step functions, i.e. to let the first elements of $w_i$ be zero and to let the elements $i, i+1, \ldots, m$ be equal to one.

Introducing the SVD of the matrix $W$

\[
W = \tilde{U} \tilde{\Sigma} \tilde{V}
\]
and assuming that \( W \) has rank \( m + 1 \) it is straightforward to derive an update equation for the coefficients \( C_k \). They are updated according to

\[
C_{k+1} = \tilde{Q}(C_k + \tilde{L}E_k^0)
\]  

(32)

where

\[
\tilde{Q} = W^+ Q W \quad \tilde{L} = W^+ L
\]

(33)

and \( W^+ \) is the pseudo-inverse of \( W \), i.e. the \((m + 1) \times N\) matrix

\[
W^+ = V \Sigma^\dagger \tilde{U}^T
\]

(34)

and \( \Sigma^\dagger \) is formed using the inverse of the singular values in \( \Sigma \). Choosing \( m < N \) the amount of computations needed to carry out the updating procedure is reduced, compared to the standard situation where the whole input vector is updated.

3 Non-minimum phase systems

Since the ultimate aim in ILC is to, when it is possible, generate a signal which is the reference signal filtered through the inverse of the system model, it is obvious that there will be problems when the system has zeros outside the unit circle. In this section this problem will be discussed. Initially the occurrence of zeros outside the unit circle in general will be discussed and then it will be shown how the optimization based ILC algorithm deals with this problem.

In e.g. (Åström and Wittenmark 1984) the following result can be found. Consider an \( n \)th order continuous time system \( G(s) \) with \( m \) zeros \( z_i, i = 1, \ldots, m \). Assuming that zero order hold is applied the corresponding discrete time system \( G_T(q) \) will have the following properties as \( T \) tends to zero:

- The discrete time system \( G_T(q) \) will have \( n - 1 \) zeros.
- \( m \) zeros of \( G_T(q) \) will be given by \( e^{z_i T} \).
- The remaining \( r = n - 1 - m \) zeros will be the zeros of a polynomial \( P_r(q) \), which for \( r = 1 \) has the zero \(-1\) and for \( r > 1 \) has zeros outside the unit circle.

This means that for a continuous time system with relative degree larger than one the discrete time system will, for short sampling intervals, have zeros on or outside the stability boundary.

It will now be shown how the design variables in optimization approach can be used to handle this situation. Recall the update equation (24)

\[
U_{k+1} = Q(U_k + LE_k^0) = QU_k + QLE_k^0
\]

(35)

with \( Q \) and \( L \) given by equations (21) and (22) respectively. When the iterations have converged the asymptotic input vector is given by

\[
U_\infty = (I - Q + QLT_n)^{-1}QLR
\]

(36)
in the general case, and

$$U_\infty = (W_u + T_u^T W_e T_u)^{-1} T_u^T W_e R$$  \hfill (37)$$

in the optimization case. In (37) it has been assumed that the nominal model equals the true system, i.e. $T_u^0 = T_u$. The properties of the algorithm will be studied by investigating the 2-norm, i.e. the largest singular value, of the matrices $Q, QL$ and $(W_u + T_u^T W_e T_u)^{-1} T_u^T W_e$. The discussion is furthermore confined to the case $W_e = I$ and $W_u = \rho \cdot I$. The following result can then be stated.

**Theorem 3.1** Consider the matrices $Q$ and $QL$ where $Q$ and $L$ are given by equations (21) and (22) in the situation $W_e = I$ and $W_u = \rho \cdot I$, where $\rho > 0$. Then

$$\|Q\|_2 < 1$$  \hfill (38)$$

$$\|QL\|_2 \leq \frac{1}{2\sqrt{\rho + \lambda}}$$  \hfill (39)$$

and

$$\|(\rho \cdot I + T_u^T T_u)^{-1} T_u^T\|_2 \leq \frac{1}{2\sqrt{\rho}}$$  \hfill (40)$$

**Proof 3.1** Introduce the SVD of the matrix $T_u$

$$T_u = \tilde{U} \Sigma \tilde{V}^T$$  \hfill (41)$$

where the singular values of $T_u$ i.e. the diagonal element of $\Sigma$ are denoted $\sigma_i$. Introducing this representation in the definitions of $Q$ and $L$ together with the assumptions for $W_u$ and $W_e$ gives

$$Q = \tilde{V}^T \Sigma^Q \tilde{V}$$  \hfill (42)$$

where the diagonal elements of $\Sigma^Q$ are given by

$$\sigma_i^Q = \frac{\sigma_i^2 + \lambda}{\sigma_i^2 + \lambda + \rho}$$  \hfill (43)$$

This is a monotonous function of $\sigma_i$ asymptotically tending to one. Similar calculations give that

$$QL = \tilde{V} \Sigma^{QL} \tilde{U}^T$$  \hfill (44)$$

where the diagonal elements $\sigma_i^{QL}$ are given by

$$\sigma_i^{QL} = \frac{\sigma_i}{\rho + \lambda + \sigma_i^2}$$  \hfill (45)$$

Maximizing the right hand side of this expression w r t $\sigma_i \geq 0$ gives that

$$\sigma_i^{QL} \leq \frac{1}{2\sqrt{\rho + \lambda}} \quad \forall \ i$$  \hfill (46)$$

Finally similar calculations give

$$(\rho \cdot I + T_u^T T_u)^{-1} T_u^T R = \tilde{V} \Sigma^\infty \tilde{U}^T$$  \hfill (47)$$

9
where the diagonal elements of $\Sigma^\infty$ are given by

$$\sigma_i^\infty = \frac{\sigma_i}{\rho + \sigma_i^2}$$  \hspace{1cm} (48)

Maximizing the right hand side gives that

$$\sigma_i^\infty \leq \frac{1}{2\sqrt{\rho}} \hspace{1cm} \forall \ i$$  \hspace{1cm} (49)

The conclusions of the discussion above are the following. When dealing with computer control of real systems it is rather likely that the corresponding discrete time model will have some zero outside the stability region. It can easily be seen using simple examples that this corresponds to very small singular values of the corresponding matrix $T_u$. The use of the input weighting in the criterion $W_u > 0$, which implies $Q \neq I$ then offers a method to control the “gain” from the error (or reference) signal to the applied input signal.

4 Frequency Domain Interpretation

The derivation and analysis of the ILC algorithm have so far been carried out in the time domain. In many cases however useful insights and interpretations of an ILC algorithm are achieved by considering the problem in the frequency domain. The aim in this section is to present a frequency domain interpretation of the ILC algorithm derived using optimization and illustrate some of its properties.

The starting point when deriving a frequency domain interpretation of the updating equation (24) is to use equation (4) in its simplest version, i.e.

$$Y = T_u U$$  \hspace{1cm} (50)

i.e. to express the filtering of a data vector as a matrix multiplication. Consider now a discrete time system with pulse response $t_i, \ l = 0, \ldots, \infty$ and transfer operator $T_u(q)$ defined by

$$T_u(q) = \sum_{i=0}^{\infty} t_i q^{-i}$$  \hspace{1cm} (51)

where $q$ denotes the shift operator. Applying an input signal $u(t)$ which is zero for $t < 0$ the output can be written

$$y(t) = \sum_{i=0}^{t} t_i u(t - l)$$  \hspace{1cm} (52)

i.e.

$$y(t) = T_u(q)u(t)$$  \hspace{1cm} (53)

Provided that the inverse exists the relationship

$$Y = T_u^{-1} U$$  \hspace{1cm} (54)
corresponds to the filtering
\[ y(t) = \frac{1}{T_u(q)}u(t) \]  
(55)

Furthermore the operation
\[ Y = T_u^T U \]  
(56)
corresponds to a non-causal filtering
\[ y(t) = t_0 u(t) + t_1 u(t + 1) + \ldots + t_{N-1} u(N) = \sum_{i=0}^{N-1} t_i u(t + l) = \sum_{i=0}^{N-1} t_i \left( \frac{1}{q} \right)^i u(t) = T_u \left( \frac{1}{q} \right) u(t) \]  
(57)
(58)

Hence the operation
\[ Y = T_u^T T_u U \]  
(59)
corresponds to
\[ y(t) = T_u(q) T_u \left( \frac{1}{q} \right) u(t) \]  
(60)
or, in the frequency domain,
\[ Y(e^{j\omega T}) = \left| T_u(e^{j\omega T}) \right|^2 U(e^{j\omega T}) \]  
(61)
i.e. a filtering operation that gives zero phase shift. One implementation of this operation is provided by the Matlab command \texttt{filtfilt}.

Now consider again equation (24)
\[ U_{k+1} = Q(U_k + LE_k^0) \]  
(62)
where putting \( W_e = I \) and \( W_u = \rho \cdot I \) the matrices \( Q \) and \( L \) are given by
\[ Q = (\rho \cdot I + \lambda \cdot I + T_u^T T_u)^{-1}(\lambda \cdot I + T_u^T T_u) \]  
(63)
and
\[ L = (\lambda \cdot I + T_u^T T_u)^{-1}T_u^T \]  
(64)

In the frequency domain equation (62) consequently corresponds to the filtering operation
\[ U_{k+1}(z) = Q(z)(U_k(z) + L(z)E_k^0(z)) \]  
(65)
where
\[ Q(z) = \frac{\lambda + T_u(z)T_u(z^{-1})}{\rho + \lambda + T_u(z)T_u(z^{-1})} \]  
(66)
and
\[ L(z) = \frac{T_u(z^{-1})}{\lambda + T_u(z)T_u(z^{-1})} \]  
(67)
Equations (66) and (67) hence represent particular choices of the filters in equation (2).

It should be observed that all boundary effects in the beginning and the end of the data sequences are neglected in the derivation of the transfer function formulation. Hence the update equations (65) and (24) do not give exactly the same results. If the length of each movement \( N \) is large the frequency domain expressions give a useful interpretation of the optimization based ILC algorithm.

Assuming that the nominal model equals the true system the convergence criterion

\[
| Q(e^{j\omega})\| 1 - L(e^{j\omega})T_u(e^{j\omega}) | < 1
\]

becomes

\[
\frac{\lambda}{\lambda + \rho + | T_u(e^{j\omega}) |^2} < 1
\]

(68)

This condition is satisfied for all \( \omega \) provided that \( \rho > 0 \). Since the gain of the system tends to zero for high frequencies the left hand side of the inequality tends to \( \lambda/(\lambda + \rho) \) for high frequencies. It is also of interest to compute the asymptotic error that is obtained provided that the ILC algorithm converges and this is given by

\[
E_\infty(z) = \frac{\rho}{\rho + T_u(z)T_u(z^{-1})}E_0(z)
\]

(70)

The value of \( \rho \) hence influences the magnitude of the error after the iterations have converged. Choosing \( \rho > 0 \) guarantees that the condition in (69) holds, but the price is that there will be an remaining error also after the iterations have converged. This is a well known fact in ILC but here this trade off is concentrated in one scalar parameter.

5 Numerical Example

In this section an example will be given where the properties of the filters \( L(z) \) and \( Q(z) \) are investigated. The example will be based on the first order discrete-time system

\[
T_u(z) = \frac{0.07z^{-1}}{1 - 0.93z^{-1}}
\]

(71)

presented in (Norrlof 1998) and obtained by identification of an ABB IRB 1400. This model will also be used in the experiments in Section 6. The data used for the identification was collected while the robot was controlled by the conventional control system. This means that \( T_u(z) \) in equation (71) is a model of the closed loop system from the ILC input signal \( u_k(t) \) to the output signal \( y_k(t) \). See also Figure 1.

The filter \( L(z) \) is determined by the nominal model \( T_u(z) \) and the parameter \( \lambda \) and in equation (67) it is seen that when \( \lambda \) decreases the filter tends towards the inverse of the nominal model. In Figure 2 and Figure 3 respectively the gain and phase of \( L(z) \) are shown for different values of \( \lambda \). For small values of \( \lambda \) the gain of \( L(z) \) is close to the gain of \( T_u^{-1}(z) \) while it for larger values obtains more low-pass character. The phase of \( L(z) \) is however not affected by \( \lambda \) and it is equal to the phase of \( T_u^{-1}(z) \).

The properties of \( Q(z) \) are determined by both \( \lambda \) and \( \rho \) and the shape of \( Q(z) \) depends on both their values and their relative size. Provided that the gain of \( T_u(z) \) decreases as \( \omega \) increases the gain of
Figure 2: Magnitude of $T_{u}^{-1}(z)$ (solid) and $L(z)$. Dashed: $\lambda = 10^{-3}$, dotted: $\lambda = 10^{-2}$, dash-dotted: $\lambda = 10^{-1}$

Figure 3: Phase of $T_{u}^{-1}(z)$ and $L(z)$. 
$Q(z)$ tends to $\lambda/(\lambda + \rho)$. This means that the value of $\rho$ has to be of the same magnitude as $\lambda$ in order to give a significant low pass behavior of $Q(z)$. When $\rho$ is essentially smaller than $\lambda$ the high frequency gain of $Q(z)$ is close to unity. This can be seen in Figure 5. The low frequency gain of $Q(z)$ does also depend on the low frequency gain of the nominal model $T_u(z)$. Some examples are given in Figure 4 and Figure 5 respectively.

![Gain of Q(z)](image)

Figure 4: Gain of $Q(z)$ for $\lambda = 10^{-3}$. Solid: $\rho = 10^{-3}$, dashed: $\rho = 10^{-2}$, dash-dotted: $\rho = 10^{-1}$
Figure 5: Gain of $Q(z)$ for $\lambda = 10^{-1}$. Solid: $\rho = 10^{-3}$, dashed: $\rho = 10^{-2}$, dash-dotted: $\rho = 10^{-1}$

6 Experimental Results

The experiments are carried out on the ABB IRB 1400 installed at the Division of Automatic Control, Linköping University. In the experiment a desired motion of axis one is specified, and the desired trajectory is given in Figure 6. It should be noticed that all signals correspond to the motor side, i.e. before the gear-box, since no measurements on the arm side are available.

Figure 6: Reference trajectory on the motor side

The nominal model required in order to design the ILC updating algorithm is obtained by using system identification and this results in the first order model, equation 71, presented in the previous section. Further details concerning this step is given in (Norrlöf 1998). Using this model the matrix $\mathbf{T}_u$ containing the impulse response coefficients is formed and then the matrices $\mathbf{Q}$ and $\mathbf{L}$ are computed.
The design variables were chosen as $\lambda = 10^{-3}$ and $\rho = 10^{-2}$. Since the updating of the ILC input signal is carried out in Matlab in the experiment equipment a straightforward implementation of the updating formula (24) is feasible. In the experiments an interface between the robot control system S4C and Matlab is used. See also (Norrlöf 1998).

Four iterations are carried out and the maximum value of the absolute value of the error in each iteration is shown in figure 7. The first value in the plot hence corresponds to the error obtained without any ILC input applied. Already after one iteration the error is almost completely eliminated, and after two iterations the level of the error corresponds to the resolution in the measurement device. Figure 8 shows the error signal at iterations 0, 1 and 2 respectively. In Figures 7 and 8 the errors are normalized.

![Figure 7: Normalized maximum absolute value of the error in each iteration.](image1)

![Figure 8: Normalized error signal. Solid: $k = 0$ (without ILC input), Dashed: $k = 1$, Dash-dotted: $k = 2$.](image2)
7 Conclusions

Design of ILC algorithms using optimization has been studied. It has been shown that optimization based ILC algorithms have an intuitive interpretation in the frequency domain and that they offer a convenient way to deal with non-minimum phase discrete-time systems. The usefulness of this kind of ILC algorithm has also been illustrated with experiments carried out on a commercial industrial robot.

References


