An adaptive approach to iterative learning control with experiments on an industrial robot

Mikael Norrlöf

Division of Automatic Control
Department of Electrical Engineering
Linköpings universitet, SE-581 83 Linköping, Sweden
WWW: <http://www.control.isy.liu.se>
Email: mino@isy.liu.se

16th August 2001

Report No.: [LiTH-ISY-R-2372](#)

Submitted to European Control Conference, ECC 2001, Sep 4-7

Technical reports from the Automatic Control group in Linköping are available by anonymous ftp at the address [ftp.control.isy.liu.se](ftp://control.isy.liu.se). This report is contained in the file 2372.pdf.

Abstract

An adaptive approach to Iterative Learning Control (ILC) based on a Kalman filter and an optimization of a quadratic criterion is presented. By estimating one of the design parameters in the Kalman filter an adaptive gain in the ILC updating formula is created. The proposed ILC design is compared with two other ILC schemes and they are all implemented on an industrial robot. The results show that the proposed adaptive ILC scheme is fast, and also robust since the gain is reduced as the error is decreased.

Keywords: Robot Application, Robot Motion Control, Sampled Data Systems, Iterative Learning Control

AN ADAPTIVE APPROACH TO ITERATIVE LEARNING CONTROL WITH EXPERIMENTS ON AN INDUSTRIAL ROBOT

M. Norrlöf

Division of Automatic Control,
Department of Electrical Engineering,
Linköping University
SE-581 83 Linköping, Sweden
fax: +46-13-282622
e-mail: mino@isy.liu.se

Keywords: Robot Application, Robot Motion Control, Sampled Data Systems, Iterative Learning Control

Abstract

An adaptive approach to Iterative Learning Control (ILC) based on a Kalman filter and an optimization of a quadratic criterion is presented. By estimating one of the design parameters in the Kalman filter an adaptive gain in the ILC updating formula is created. The proposed ILC design is compared with two other ILC schemes and they are all implemented on an industrial robot. The results show that the proposed adaptive ILC scheme is fast, and also robust since the gain is reduced as the error is decreased.

1 Introduction

Iterative Learning Control is now a well established method for control of repetitive processes. In general it is considered to be an approach for trajectory tracking and this is how it is usually described in the literature, see e.g., the surveys [12, 13, 4]. In this paper we will use ILC in a different setting where it is applied for disturbance rejection¹. ILC has been used in this framework earlier, see for example [17, 6, 5], where disturbances such as initial state disturbances and measurement disturbances are discussed.

In Section 3 we will show how we can apply the results in a standard tracking application for ILC. In Figure 1 the structure used in the disturbance rejection formulation approach to ILC is shown as a block diagram.

Figure 1: The system structure considered in the disturbance rejection formulation of the ILC problem.

The goal in ILC is usually to, iteratively, find the input to a

system such that some error is minimized. In the disturbance rejection formulation, the goal becomes to find an input $u_k(t)$ such that the output $z_k(t)$ is minimized. If the system is known and invertible, and the disturbance $d_k(t)$ is known, the obvious approach would be to filter $d_k(t)$ through the inverse of the system and then use the resulting $u_k(t)$ as a control input. This means that the optimal input looks like,

$$u_k(t) = -(G^0)^{-1}d_k(t)$$

Different aspects of the disturbance rejection approach to ILC will be considered in this paper. Results from using the methods on an industrial robot will also be presented.

2 A State Space Based Approach to ILC

The structure of the system in the disturbance rejection formulation of ILC is shown in Figure 1. Next the mathematical system description will be discussed, after that the adaptive ILC method is introduced.

2.1 Matrix description of the system

An ILC system is characterized by the fact that it is only defined over a finite interval of time. If the sampling time is equal to one, this means that $0 \leq t \leq n-1$. This is also the reason why it is possible to describe the system in a matrix form,

$$\begin{aligned} z_k &= G^0 u_k + d_k \\ y_k &= z_k + n_k \end{aligned} \quad (1)$$

with

$$d_{k+1} = d_k + \Delta_{d_k} \quad (2)$$

where $z_k, u_k, d_k, y_k, n_k, \Delta_{d_k} \in \mathbb{R}^n$ and $G^0 \in \mathbb{R}^{n \times n}$. The vectors are defined as,

$$z_k = [z_k(0) \quad z_k(1) \quad \dots \quad z_k(n-1)]^T$$

with the other vectors defined accordingly. The matrix G^0 is for a causal system a lower triangular matrix and if the system is linear time invariant the matrix G^0 also becomes Toeplitz. This particular structure of ILC systems has been exploited earlier in for example, the work by Moore [12, 14], and also in Phan et al. [19] and Lee et al. [9].

¹More details on the disturbance rejection formulation can be found in [15].

We assume that $\mathbf{d}_k$ and $\mathbf{n}_k$ are random with covariance matrices for Δ_{d_k} and $\mathbf{n}_k$ given by $\mathbf{R}_{\Delta_{d,k}}$ and $\mathbf{R}_{n,k}$ respectively. Further more it is assumed that $\mathbf{n}_k(t) = \nu_n(\bar{t})$ and $\Delta_{d_k}(t) = \nu_{\Delta}(\bar{t})$ with $\bar{t} = k \cdot t$ and $\nu_{(\cdot)}$ a white stationary stochastic process.

Another representation of the system in (1) that can be useful is

$$\begin{aligned} z_k(t) &= G^0(q)u_k(t) + d_k(t) \\ y_k(t) &= z_k(t) + n_k(t) \end{aligned} \quad (3)$$

with

$$d_{k+1}(t) = d_k(t) + \Delta_{d_k}(t) \quad (4)$$

Note that the description in (1) is more general than (3) since it also covers linear time variant systems. When considering linear time invariant systems, however, the two representations are equivalent.

Assume that the system description is not perfectly known, but instead a model $\mathbf{G}$ is used and

$$\mathbf{G}^0 = \mathbf{G}(\mathbf{I} + \Delta_G) \quad (5)$$

where Δ_G is a relative model uncertainty. Now, using the updating formula for the disturbance, $\mathbf{d}_k$, in (4) and the system description from (5), it is possible to write (1) in the following form

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{z}_k - \mathbf{G}(\mathbf{u}_{k+1} - \mathbf{u}_k) - \mathbf{G}\Delta_G(\mathbf{u}_{k+1} - \mathbf{u}_k) + \Delta_{d_k} \\ \mathbf{y}_k &= \mathbf{z}_k + \mathbf{n}_k \end{aligned} \quad (6)$$

The last two terms in the first equation can be considered as disturbances since they are both unknown. It is however known that the first one depends on the difference between two consecutive control signals. If the model uncertainty is small and/or the updating speed of the control signal is slow, this disturbance will have a small effect on the resulting system.

2.2 Estimation procedure

A linear estimator for the system described in (6) can be written as

$$\hat{\mathbf{z}}_{k+1} = \hat{\mathbf{z}}_k - \mathbf{G}(\mathbf{u}_{k+1} - \mathbf{u}_k) + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{z}}_k) \quad (7)$$

where $\mathbf{K}_k$ is the gain of the estimator. By applying standard Kalman filter techniques, see for example [3], the estimation procedure for $\hat{\mathbf{z}}_k$ becomes

$$\hat{\mathbf{z}}_{k+1} = \hat{\mathbf{z}}_k - \mathbf{G}(\mathbf{u}_{k+1} - \mathbf{u}_k) + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{z}}_k) \quad (8a)$$

$$\mathbf{K}_k = \mathbf{P}_k(\mathbf{P}_k + \hat{\mathbf{R}}_{n,k})^{-1} \quad (8b)$$

$$\mathbf{P}_{k+1} = \mathbf{P}_k + \hat{\mathbf{R}}_{\Delta_{d,k}} - \mathbf{P}_k(\mathbf{P}_k + \hat{\mathbf{R}}_{n,k})^{-1}\mathbf{P}_k \quad (8c)$$

where it is assumed that Δ_{d_k} and $\mathbf{n}_k$ are uncorrelated. Compare also the discussion in the beginning of Section 2.1.

2.3 An optimization based approach to ILC

Consider the following criterion for control of (1),

$$J_k = \mathbf{z}_k^T \mathbf{W}_z \mathbf{z}_k + \mathbf{u}_k^T \mathbf{W}_u \mathbf{u}_k \quad (9)$$

By minimizing (9) it is possible to find an optimal input to the system, with respect to the criterion. This has been studied in for example [4, 2, 1, 18, 7, 9] but in contrast to most of the approaches in the literature, the term containing $\mathbf{u}_k - \mathbf{u}_{k-1}$ is not included here.

By using, in (9), the definition of $\mathbf{z}_k$ from (3) and taking the derivative with respect to $\mathbf{u}_k$ it follows that

$$\frac{\partial J_k}{\partial \mathbf{u}_k} = ((\mathbf{G}^0)^T \mathbf{W}_z \mathbf{G}^0 + \mathbf{W}_u) \mathbf{u}_k + (\mathbf{G}^0)^T \mathbf{W}_z \mathbf{d}_k$$

Now solve for $\mathbf{u}_k$ when $\frac{\partial J_k}{\partial \mathbf{u}_k} = 0$. This leads to

$$\mathbf{u}_{k+1}^* = -((\mathbf{G}^0)^T \mathbf{W}_z \mathbf{G}^0 + \mathbf{W}_u)^{-1} (\mathbf{G}^0)^T \mathbf{W}_z \mathbf{d}_{k+1} \quad (10)$$

where the $*$ denotes the optimal input.

If $\mathbf{W}_u = 0$ and $\mathbf{d}_{k+1}$ are known, then the updating scheme for the control $\mathbf{u}_k$ becomes

$$\mathbf{u}_{k+1}^* = -(\mathbf{G}^0)^{-1} \mathbf{d}_{k+1} \quad (11)$$

which we recognize from Section 1. Note that this expression actually contains a feedforward from the disturbance $\mathbf{d}_{k+1}$. From a practical point of view (11) is not very useful since when $\mathbf{u}_{k+1}$ is calculated, $\mathbf{d}_{k+1}$ is in general not available. If $\mathbf{d}$ does not change as a function of iteration it will however work since old estimates of $\mathbf{d}$ can be used. In practice the control solution can, of course, not use the true system description. If instead a model of the system is used the control signal $\mathbf{u}_k$ can be calculated as

$$\mathbf{u}_{k+1} = -(\mathbf{G}^T \mathbf{W}_z \mathbf{G} + \mathbf{W}_u)^{-1} \mathbf{G}^T \mathbf{W}_z \hat{\mathbf{d}}_{k+1} \quad (12)$$

In (12) it is also taken into account that the true $\mathbf{d}_{k+1}$ is not available directly as a measured signal. An estimate of $\mathbf{d}_{k+1}$ is found as

$$\hat{\mathbf{d}}_{k+1} = \hat{\mathbf{z}}_{k+1} - \mathbf{G} \mathbf{u}_{k+1} \quad (13)$$

which means that the expression for $\mathbf{u}_k$ can be simplified to

$$\mathbf{u}_{k+1} = -\mathbf{W}_u^{-1} \mathbf{G}^T \mathbf{W}_z \hat{\mathbf{z}}_{k+1} \quad (14)$$

using (12) and (13). This can be plugged into the observer in (7) resulting in

$$\hat{\mathbf{z}}_{k+1} = \hat{\mathbf{z}}_k + (\mathbf{I} + \mathbf{G} \mathbf{W}_u^{-1} \mathbf{G}^T \mathbf{W}_z)^{-1} \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{z}}_k) \quad (15)$$

Together with (14) and the calculation of $\mathbf{K}_k$ from the previous section, this gives an ILC scheme with two iterative updating formulas, including the one for $\mathbf{P}_k$. Compared to the traditional ILC schemes,

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

this means that the iterative behavior of the ILC algorithm has moved from the updating of the control signal to the estimator.

2.4 An adaptive algorithm for ILC

The calculations of P_k and K_k in the time varying Kalman filter in Section 2.2 do not include the measurements from the system. In this section a possible extension to the algorithm presented in the previous sections is given. The new algorithm takes advantage of the measurements from the system and use them to adapt a measure of the variability of the system disturbance, $\hat{R}_{\Delta,k}$. The algorithm is adaptive since the value of K_k will depend on the variability measure through P_k .

To explain the idea behind the measure of variability used in the algorithm first note that the system model G does not capture the true system dynamics perfectly. Instead the relation given by (6) describes the true system in terms of the model and the uncertainty.

The idea is to use

$$z_{k+1} = z_k - G(u_{k+1} - u_k) - \underbrace{G\Delta_G(u_{k+1} - u_k) + \Delta_{d_k}}_{\Delta}$$

and find a measure of the size of the variation of Δ . The following equation serves this purpose

$$\hat{r}_{\Delta,k} = \frac{1}{n-1} (u_{k+1} - u_k)^T \hat{\Delta}_G^T G^T G \hat{\Delta}_G (u_{k+1} - u_k) + \hat{r}_{\Delta_d} \quad (16)$$

where $\hat{\Delta}_G$ is an estimate of the true model uncertainty and $\hat{r}_{\Delta_d}$ is an estimate of the variance of Δ_{d_k} . The resulting ILC algorithm can now be formulated.

Algorithm 1 (Adaptive optimization based ILC algorithm)

1. Design an ILC updating equation using the LQ design in Section 2.3.
2. Assume $\hat{R}_{\Delta_d}$ and $\hat{R}_n$ diagonal with the diagonal elements equal to $\hat{r}_{\Delta_d}$ and $\hat{r}_n$ respectively, i.e., $\hat{R}_{\Delta_d} = \hat{r}_{\Delta_d} \cdot I$ and $\hat{R}_n = \hat{r}_n \cdot I$. Choose $\hat{r}_{\Delta_d}$ and $\hat{r}_n$ from physical insight or such that p_∞ and the corresponding κ_∞ get the desired values.
3. Let $\hat{z}_0 = 0$.
4. Choose an initial value for p_0 . This can be a large number since it will converge to $p_1 \approx \hat{r}_n + \hat{r}_{\Delta_d}$ already after one iteration.
5. Implementation of the ILC algorithm:

(a) Let $k = 0$, and $u_0 = W_u^{-1} G^T W_z \hat{z}_0$.

(b) Apply u_k and measure y_k .

(c) Calculate,

$$\begin{aligned} \kappa_k &= \frac{p_k}{p_k + \hat{r}_n} \\ \hat{z}_{k+1} &= \hat{z}_k + (I + G W_u^{-1} G^T W_z)^{-1} \kappa_k (y_k - \hat{z}_k) \\ u_{k+1} &= W_u^{-1} G^T W_z \hat{z}_{k+1} \\ \hat{r}_{\Delta,k} &= \frac{1}{n-1} (u_{k+1} - u_k)^T \hat{\Delta}_G^T G^T G \hat{\Delta}_G (u_{k+1} - u_k) + \hat{r}_{\Delta_d} \\ &\quad G \hat{\Delta}_G (u_{k+1} - u_k) + \hat{r}_{\Delta_d} \\ p_{k+1} &= \frac{p_k \hat{r}_n}{p_k + \hat{r}_n} + \hat{r}_{\Delta,k} \end{aligned}$$

(d) Let $k = k + 1$. Start again from (b).

It is important to understand the properties of the proposed algorithm and this, especially, includes stability and performance. More details on these properties are presented in [15]. The main result is to show boundedness of the estimate $\hat{z}_k$ and this implies that the resulting ILC algorithm is stable. The analysis in [15] is limited to some particular assumptions on the uncertainty in the system.

The idea of using an optimization based ILC updating equation and an estimation procedure is also covered in Chapter 9 (written by Lee and Lee) of [4]. There solution does not have the same criterion in the control design and the observer is not adaptive as is the case here. Adaptive ILC algorithms are also covered in, e.g., [8, 20, 16]. Notice that many proposed adaptive ILC algorithms are combination of adaptive feedback controllers and ILC algorithms. The adaptive ILC algorithm presented in this paper is instead truly adaptive and, it does not say anything about the feedback control solution of the system.

Figure 2: The ABB IRB1400 manipulator.

3 Description of the Experiment

The industrial robot used in the experiment is shown in Figure 2 and ILC is applied to 3 of the total 6 joints of the robot. Each joint is modeled as a transfer operator description from the ILC control input to the measured motor position of the robot, i.e., G^0 in (3). For a complete description of this procedure the

Figure 3: The program used to produce the trajectory used in the example (upper) and the resulting trajectory on the arm-side translated such that p1 is in the origin (lower).

reader is referred to [15] but here we will give a short description. First we note that the problem we get when controlling the robot is a classical ILC tracking problem. In Figure 4 the configuration of the system considered in the experiments is shown. Clearly this is different from the structure of the stan-

Figure 4: The system configuration used in the experiments.

dard system description in the disturbance rejection approach from Figure 1. The reference signal is one of the inputs to the system and the goal of the control now becomes to track the reference trajectory $r(t)$. By using the reference signal $r(t)$ as the disturbance we get the control variable $z_k(t)$ as the control error that we want to minimize. It is now straightforward to apply the algorithm presented in Section 2.4 to the problem. Note that in the application the parameter $\hat{r}_{\Delta_d}$ does not have a direct interpretation as a physical parameter. The parameter $\hat{r}_n$ however, still have a physical meaning and can be chosen accordingly. From Algorithm 1 we also know that $\hat{r}_{\Delta_d}$ can be chosen such that p_∞ and κ_∞ get the desired values and this is the approach taken here. The limit value, p_∞ , is found by solving

$$p_\infty = \frac{p_\infty \hat{r}_n}{p_\infty + \hat{r}_n} + \hat{r}_{\Delta_d}$$

which gives

$$p_\infty = \frac{\hat{r}_{\Delta_d}}{2} \left(1 + \sqrt{1 + 2 \frac{\hat{r}_n}{\hat{r}_{\Delta_d}}} \right) \quad (17)$$

From the definition of κ_k it is then straightforward to compute κ_∞ .

The model G that we need for the design and implementation of the ILC scheme is found by using the setup in Figure 4 and exciting the system with an input u while measuring the output y . Based on these measurements a linear time invariant discrete time black-box model is computed using *System Identification Toolbox* in MATLABTM [10]. One model is found for each joint giving a total of 3 models. The modeling process can also give some indication on the model error. One possible approach is the model error model method, see for example [11], which gives a measure of the model error which then can be translated into Δ_G .

To evaluate the proposed adaptive ILC algorithm the program shown in Figure 3 is used. In Figure 3 the resulting desired trajectory on the arm-side of the robot is also shown.

To make it possible to rank the adaptive ILC algorithm, two different algorithms have been chosen for comparison. The first is a traditional ILC algorithm formulation with the updating scheme given by

$$u_{k+1}(t) = Q(q)(u_k(t) + L(q)y_k(t)) \quad (18)$$

The second algorithm is the same as the adaptive ILC algorithm, except that the Kalman gain κ_k is fixed to a value slightly less than one ($\kappa_k = 0.99$). The second case is made to show the advantage of having an adaptive gain in the updating formula.

The matrix W_u^{-1} is chosen as a realization of a low-pass zero-phase filter. This choice give higher weight to high frequencies in the control design and there it helps to make the ILC algorithm more robust [15]. For the other parameters the following values were used in the experiment,

$$\begin{aligned} p_0 &= 10^4 \\ W_z &= 10^3 \cdot I \\ \hat{r}_{\Delta_d} &= 10^{-6} \\ \hat{r}_n &= 5 \cdot 10^{-5} \\ \Delta_G &= 0.5 \cdot I \end{aligned}$$

This means that p_∞ becomes equal to $5.5 \cdot 10^{-6}$ and the corresponding κ_∞ becomes $\kappa_\infty = 0.10$ which is a reasonable lower limit for the gain κ_k . The model error is in this case not based on the identification results but instead chosen such that it represents a 50% model error.

The filters in the traditional ILC algorithm, given by (18), are chosen such that $Q(q)$ is a second order Butterworth filter with cut-off frequency 0.2 of the Nyquist frequency and $L(q) = 0.9q^4$. This choice of L -filter is based on the model that we used for the design of the adaptive ILC algorithm and it gives good robustness properties [15].

4 Results

Using the experiment described in Section 3, the three ILC designs are tested on the robot. The results from the experiments are evaluated on the motor side of the robot. This is also where the measurements and the control are performed. For an industrial application the arm side position is more interesting to consider since it is the actual position of the tool that the customer programs and that he wants to follow a programmed trajectory. An evaluation of the result on the arm side is not considered here, instead this is left for future work.

The results on the motor-side from the experiments with the three ILC algorithms are shown in Figure 5. Note that the measures in Figure 5 are normalized and that both the ∞ -norm as well as the 2-norm are depicted.

Clearly, the transient response of the learning is better with the adaptive ILC scheme. The ILC algorithm designed according to the adaptive ILC scheme but with the Kalman gain kept constant is, however, not so robust which is shown by the fact that $\|y_k\|_2$ for motor 1 actually starts growing after about 7 iterations. In Figure 6 the values of the gain, κ_k , in the adaptive ILC algorithms are shown as a function of iteration. They are large in the first iterations where the control error is still big, but as it slowly vanishes the gain also decreases. Note that in addition to what is said in Algorithm 1, it is important to choose the correct size of $\hat{r}_{\Delta_d}$ in order to get this effect. If $\hat{r}_{\Delta_d}$ is chosen too large this value will dominate $\hat{r}_{\Delta,k}$ and then κ_k will not decrease as shown in Figure 6, instead it will decrease like $\frac{1}{k+1}$.

5 Conclusions

Using a disturbance rejection formulation of ILC as it is presented in [15] a new ILC algorithm has been developed. The basic idea is to introduce an iteration varying gain in the ILC procedure in order to be more robust against model errors but also to reduce the impact of measurement noise.

Results from state space modeling and design are used to create the ILC algorithm. The proposed algorithm works also when the system is not perfectly known. The design is based on an LQ-solution and a time variable Kalman filter where one of the design variables in the Kalman filter is calculated from data. This means that the algorithm is, in fact, adaptive.

Experiments with the proposed adaptive algorithm applied to an industrial robot are made. The results show an improvement in the tracking on the motor-side of the robot and the proposed adaptive and model based ILC algorithm is shown to give better results than a traditional ILC algorithm with constant gain. It is also shown that by reducing the gain a more robust solution is achieved since the size of the errors tends to grow when the gain is forced to be constant in the adaptive ILC scheme.

Figure 5: The normalized ∞ -norm and 2-norm of the error for the different ILC algorithms. The adaptive ILC scheme ($\times$), the adaptive scheme with κ_k constant ($\circ$), and the traditional ILC scheme given by (18) ($\diamond$).

Figure 6: The value of κ_k for the ILC associated with the three different motors.

Acknowledgment

This work was supported by NUTEK's Center of Excellence ISIS at Linköping University, Linköping, Sweden.

References

- [1] N. Amann, D. H. Owens, and E. Rogers. Iterative learning control for discrete time systems with exponential rate of convergence. Technical Report 95/14, Centre for Systems and Control Engineering, University of Exeter, 1995.
- [2] N. Amann, D. H. Owens, and E. Rogers. Iterative learning control using optimal feedback and feedforward actions. Technical Report 95/13, Centre for Systems and Control Engineering, University of Exeter, 1995.
- [3] Brian D.O. Anderson and John B. Moore. *Optimal Filtering*. Prentice-Hall, 1979.
- [4] Z. Bien and J.-X. Xu. *Iterative Learning Control: Analysis, Design, Integration and Application*. Kluwer Academic Publishers, 1998.
- [5] Y. Chen, C. Wen, J.-X. Xu, and M. Sun. An initial state learning method for iterative learning control of uncertain time-varying systems. In *Proc. of the 35th Conf. on Decision and Control*, pages 3996–4001, Kobe, Japan, Dec 1996.
- [6] C.-J. Chien. A discrete iterative learning control of nonlinear time-varying systems. In *Proc. of the 35th IEEE Conf. on Decision and Control*, pages 3056–3061, Kobe, Japan, Dec 1996.
- [7] S. Gunnarsson and M. Norrlöf. Some aspects of an optimization approach to iterative learning control. In *Proc. of the 38th IEEE Conference on Decision and Control*, Phoenix, Arizona, USA, Dec 1999.
- [8] T. Kuc and J. S. Lee. An adaptive learning control of uncertain robotic systems. In *Proc. of the 30th Conf. on Decision and Control*, pages 1206–1211, Brighton, England, Dec 1991.
- [9] Jay. H. Lee, Kwang S. Lee, and Won C. Kim. Model-based iterative learning control with a quadratic criterion for time-varying linear systems. *Automatica*, 36(5):641–657, May 2000.
- [10] L. Ljung. *System Identification Toolbox - For Use with Matlab*. The MathWorks Inc., 1995.
- [11] L. Ljung. Model error modeling and control design. In *The IFAC Symposium on System Identification, SYSID2000*, 2000.
- [12] K. L. Moore. *Iterative Learning Control for Deterministic Systems*. Advances in Industrial Control. Springer-Verlag, 1993.
- [13] K. L. Moore. Iterative learning control - an expository overview. *Applied and Computational Controls, Signal Processing and Circuits*, 1, 1998.
- [14] Kevin L. Moore. Multi-loop control approach to designing iterative learning controllers. In *Proc. of the 37th IEEE Conference on Decision and Control*, Tampa, Florida, USA, Dec 1998.
- [15] M. Norrlöf. *Iterative Learning Control: Analysis, Design, and Experiments*. PhD thesis, Linköping University, Linköping, Sweden, 2000. Linköping Studies in Science and Technology. Dissertations No. 653. Download from <http://www.control.isy.liu.se/publications/>.
- [16] D.H. Owens and G. Munde. Error convergence in an adaptive iterative learning controller. *International Journal of Control*, 73(10):851–857, 2000.
- [17] S. Panzieri and G. Ulivi. Disturbance rejection of iterative learning control applied to trajectory tracking for a flexible manipulator. In *Proceedings of 3rd European Control Conference*, pages 2374–2379, Sep 1995.
- [18] J.A Frueh M.Q. Phan. Linear quadratic optimal learning control (LQL). In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 678–683, Tampa, Florida, USA, 1998.
- [19] M. Phan and J.A Frueh. Learning control for trajectory tracking using basis functions. In *Proc. of the 35th IEEE Conf. on Decision and Control*, pages 2490–2492, Kobe, Japan, Dec 1996.
- [20] J-X. Xu and B. Viswanathan. Adaptive robust iterative learning control with dead zone scheme. *Automatica*, 36(1):91–99, 2000.