Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

An Optimization-based Approach for Design of Iterative Learning Controllers with Accelerated Rates of Convergence

Sandipan Mishra[1], and Masayoshi Tomizuka[2], *Fellow, IEEE.*

***Abstract*—In this paper, a new technique for designing iterative learning controllers has been proposed. The control update law is based on the minimization of a quadratic cost function. The control input update law is time varying. It is shown that the proposed controller has monotonic super-linear convergence. A systematic robustness and performance analysis has been presented to evaluate the effectiveness of the controller. The effect of different design parameters on the closed loop system performance, robustness, learning rate is investigated. The relationship between three critical indices for evaluation of ILC's – performance, rate of learning and robustness – has been studied and inferences drawn about the trade-offs. Numerical simulations verify the results.**

I. Introduction

ITERATIVE Learning Control has been extensively used in control of systems that are repetitive. When a process is repeatedly executed, it is natural to use information obtained from previous trials about the process to improve performance in the current trial. It is akin to the human "learning" pattern, hence the name Iterative Learning Control (ILC). The first paper on ILC was published by Uchiyama [1]. It was followed by similar work by Arimoto *et al* [2], Bartolini *et al* [3] and Craig [4], among others. Since then, several research efforts have been directed at design of more efficient learning algorithms [12][13][14]. In the industrial scenario, numerous applications of ILC have been explored, especially in robotics [3][12] and, of-late, in semi-conductor manufacturing [9][19]. Many books and in-depth reviews have also been published on ILC research [17][18].

Several design techniques have been proposed for ILC based controllers. One of the most effective design techniques for learning filters is based on optimality criteria. Yamakita *et al* [16] used a steepest descent method to minimize the L_2 norm of the tracking error. Phan *et al* [10] used the term Linear Quadratic Optimal Learning Control (LQL) to describe the ILC counterpart of optimal linear quadratic regulator (LQR) problems in standard control system design, and used Riccati equations to design the learning filters. Amann *et al* [6][7][8] proposed an ILC design strategy based on optimization of a quadratic performance index.

For analysis of the performance of ILC controllers, a lifted formulation of the closed loop system was first developed by Phan *et al* [20]. It provides a powerful tool for systematic analysis of learning controllers. A *synthesis* tool for learning controllers based on the lifted ILC formulation was proposed by Bosgra *et al* [21]. Recently, Feng *et al* [11] have suggested the use of a parameter optimal iterative learning controller.

One of the key issues in learning control is the "rate of learning". The rate of learning is a direct consequence of the rate of convergence of the output to the steady state output. Using the lifted ILC formulation, it is possible to get expressions for bounds on the rate of convergence of the learning algorithm. Although several interesting optimization-based learning controllers have been implemented and analyzed, the convergence guaranteed is at best linear [5][10]. There is, therefore, strong motivation for designing learning controllers with better learning behavior, i.e., with faster rates of convergence.

In this paper, we propose a new convergent iterative learning controller based on the optimization of a different quadratic cost function. We exploit the structure of the optimization problem to obtain *time varying* learning filters. The solution of the optimization problem yields a recursive relation between the learning filters from one cycle to the next. Further, we prove that the rate of convergence is faster than the standard quadratic learning controller. The notion of *super-linear convergence* is introduced. However, the controller designed by naively optimizing the cost function has poor robustness. To improve robustness while retaining super-linear convergence, the control update law mentioned above is modified slightly. A comparison of the performance of different optimization based ILC's highlights the three-way trade-off between *performance*, *robustness* and *rate of learning*.

The paper has been organized as follows. Section II introduces some nomenclature. Section III poses the iterative learning controller design problem using a lifted ILC

Manuscript received March 7th, 2005. This work was supported in part by the University of California Semiconductor Manufacturing Research and Technology Grant and NIKON Research Center of America, Belmont.

1. Sandipan Mishra is a graduate student in Mechanical Engineering at the University of California at Berkeley, Berkeley, CA 94720 USA. (phone: 510-7106507; e-mail: sandipan@ me.berkeley.edu).

2. Masayoshi Tomizuka is the Cheryl and Neerhout Jr. Distinguished Professor in the department of Mechanical Engineering at the University of California at Berkeley, Berkeley, CA 94720 USA. (e-mail: tomizuka@me.berkeley.edu)

0-7803-9568-9/05/$20.00 ©2005 IEEE

formulation. In section IV, we examine and compare two quadratic optimal learning controllers based on different optimization of different cost functions and derive expressions for convergence rates, and steady state errors. In section V, the robustness of the proposed controllers has been analyzed. Section VI proposes a scheme based on the modified Quadratic Iterative Learning Control scheme with better robustness. Finally simulation results are shown in section VII. To conclude, the effectiveness of the proposed controller is evaluated and a discussion of tradeoffs involved in quadratic cost optimal ILC filter design is presented in section VIII.

II. NOMENCLATURE

T_u – *Lifted Plant Matrix*
$g(i)$ – i^{th} *Markov Parameter (Impulse Response Coefficient)*
N – *Period of the Repetitive Process*
$(\bullet)_k(j)$ – $(\bullet)(kN+j)$
$\mathbf{u_k}$ – *Lifted Input Vector at the* k^{th} *cycle*
$\mathbf{r}$ – *Lifted Reference Vector*
$\mathbf{y_k}$ – *Lifted Plant Output Vector at the* k^{th} *cycle*
$\mathbf{e_k}$ – *Lifted Output Error Vector at the* k^{th} *cycle*
$\|\bullet\|$ – *Standard 2-norm on* $\Re^N$ *, or induced 2-norm on* $\Re^{NxN}$
$\rho(\bullet)$ – *Spectral Radius of the Matrix* $\bullet$
$\sigma(\bullet)$ – *Largest Singular Value of the Matrix* $\bullet$

III. PROBLEM FORMULATION

Lifted Formulation of Quadratic Iterative Learning Control Problem (Q-ILC)

In this paper, we will consider the application of learning controllers to stable single input single output (SISO) systems in discrete time. The plant dynamics can then be described by: $\mathbf{y}_k = T_u\mathbf{u}_k + T_r\mathbf{r}$

$$(1)$$

$$\text{where, } \begin{cases} \mathbf{y}_k = [y_k(0),\cdots,y_k(N-1)]^T \in \Re^N \\ \mathbf{u}_k = [u_k(0),\cdots,u_k(N-1)]^T \in \Re^N \\ \mathbf{r} = [r(0),\cdots,r(N-1)]^T \in \Re^N \end{cases} \tag{2}$$

Without loss of generality, we can assume $T_r = 0$, and, if the plant is LTI,

$$T_u = \begin{bmatrix} g(0) & 0 & \dots & 0 \\ g(1) & g(0) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g(N-1) & \dots & \dots & g(0) \end{bmatrix} \in \Re^{N\times N} \tag{3}$$

Remark (1): It is interesting to note that the lifted ILC formulation implicitly assumes that the initial condition at the beginning of each cycle is zero. If the initial rest condition is violated, then the plant model assumed is only an approximation of the actual plant.

The ILC problem considered here is a first order ILC design problem, which uses an update law of the form:

$$\mathbf{u}_{k+1} = Q(\mathbf{u}_k + L\mathbf{e}_k) \tag{4}$$

where the bold face letters represent the lifted super-vectors. We use the error and the control inputs in the previous cycle to compute an updated control input for the current cycle. This has been shown in a block diagram in Fig. 1.

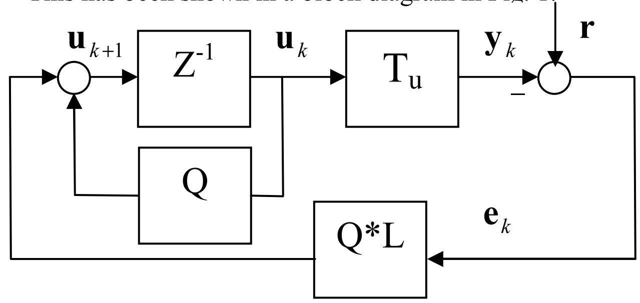

Fig. 1. Block Diagram of the Closed Loop System with the ILC. Note the internal model Z^{-1} in the controller.

It can be shown easily that [5] with the update law satisfying eq. (4), the closed loop system is stable iff $\rho(Q(I-LT_u))<1$. Further, monotonic exponential convergence is guaranteed if $\sigma(Q(I-LT_u)) = \lambda < 1$,

$$(5)$$

and the rate of convergence is governed by the following equation: $\|\mathbf{u}_\infty - \mathbf{u}_k\| \le \lambda^k \|\mathbf{u}_\infty - \mathbf{u}_0\|$ (6)

From eq. (6), with time invariant Q, and L, at best linear exponential convergence is guaranteed, if eq. (5) is satisfied. It is, however, important to notice that the final steady state input $\mathbf{u}_\infty$ need not result in perfect tracking. The final steady state value of the input vector is a key indicator for the evaluation of performance of the learning controller, since it is related directly to the steady state tracking error.

The Quadratic ILC problem can then be stated as:

> Given a reference trajectory $\mathbf{r}$, and a quadratic cost function J_k for a cycle, the objective is to design the learning matrices Q, L such that the cost function J_k is minimized, i.e.,
>
> $$\mathbf{u}_{k+1} = \arg\min\{J_{k+1}\}$$
>
> where $\mathbf{u}_{k+1} = Q(\mathbf{u}_k + L\mathbf{e}_k)$ (QILC)

IV. THE QUADRATIC OPTIMAL LEARNING CONTROL PROBLEM

In this section, the solution of the Q-ILC problem for two specific cost functions is presented. The first problem was solved by Phan *et al* [10] and Amann *et al* [6]. This is the "standard" Q-ILC problem [5]. The second problem considers a modified Q-ILC using a cleverly designed cost function.

A. Solution of the "Standard" Q-ILC Problem

Consider the problem (QILC) with the following cost function:

$$J_{k+1} = \mathbf{e}_{k+1}^T W_e \mathbf{e}_{k+1} + \mathbf{u}_{k+1}^T W_u \mathbf{u}_{k+1} + (\mathbf{u}_{k+1} - \mathbf{u}_k)^T (\mathbf{u}_{k+1} - \mathbf{u}_k) \quad (7)$$

To obtain the control update law that minimizes this cost function, we find,

$$\boxed{\left.\begin{aligned} \mathbf{u}_{k+1} &= Q(\mathbf{u}_k + L\mathbf{e}_k) \\ Q &= (I + W_u + T_u^T W_e T_u)^{-1}(I + T_u^T W_e T_u) \\ L &= (I + T_u^T W_e T_u)^{-1}(T_u^T W_e) \end{aligned}\right\} \quad (8)}$$

Remark (A1): It is interesting to observe that the matrices Q L remain the same from cycle to cycle. The advantage of this scheme is that the matrices Q and L are pre-computable offline.

***Lemma IV.A.1*:** The closed loop system using the Q-ILC controller designed above is stable.

Proof: To check for stability of the closed loop system, using eq. (5),

$$\begin{aligned} \rho(Q(I - LT_u)) &= \rho((I + W_u + T_u^T W_e T_u)^{-1}) \\ &\le \sigma((I + W_u + T_u^T W_e T_u)^{-1}) = \beta \end{aligned} \quad (9)$$

Since $W_e, W_u > 0$, $\beta < 1$. Therefore the closed loop system is stable. ◊

Lemma IV.A.2: The convergence rate of the Q-ILC is given

$$\|\mathbf{u}_k - \mathbf{u}_\infty\|_2 \le \left(\prod_{i=1}^{k} \sigma(Q_i(I - L_i T_u))\right)\|\mathbf{u}_0 - \mathbf{u}_\infty\|_2 = \beta^k \|\mathbf{u}_0 - \mathbf{u}_\infty\|_2 \quad (10a)$$

Remark (A2): From Lemmas IV.A.1-2, we get better stability and faster convergence rates if we have β<<1. This can be achieved by making W_u and W_e "larger", i.e., if $\underline{\sigma}(W_u), \underline{\sigma}(W_e) >> 0$.

Remark (A3): It is also important to observe that the rate of convergence is linear exponential. In other words, the exponent of the contraction factor β is linear in 'k'.

We now propose a lemma to evaluate the performance of the controller in terms of the steady state tracking error $\mathbf{e}_\infty$.

***Lemma IV.A.3**:* The final steady state error $\mathbf{e}_\infty$ is given by the expression

$$\begin{aligned} \mathbf{e}_\infty &= (I - T_u[W_u + T_u^T W_e T_u]^{-1} T_u^T W_e)\mathbf{r} \\ &= (I + T_u W_u^{-1} T_u^T W_e)^{-1}\mathbf{r} \end{aligned} \quad (11)$$

Remark (A4): It is clear from eq. (11) that for ensuring small steady state error, we must choose W_u as "small" as possible! Therefore, there is a tradeoff involved in choosing tracking performance vs. rate of learning.

B. Solution of a "Modified" Q-ILC Problem

In the previous section, the cost function J_{k+1} (eq.(7)) penalizes the current cycle error, the current cycle control effort, and finally, the rate of change of input from cycle to cycle. Instead of penalizing the input from cycle to cycle, it is more intuitively consistent to penalize the rate of change of the 'predicted input'. Further, we have assumed Q and L to be time invariant matrices. In this analysis, we will not restrict ourselves to choosing Q and L as time invariant matrices. So, the control update law from cycle to cycle is:

$$\mathbf{u}_{k+1} = Q_{k+1}(\mathbf{u}_k + L_{k+1}\mathbf{e}_k), \quad (12)$$

and, the "predicted control input" for the k+1st cycle is defined as: $\hat{\mathbf{u}}_{k+1} = Q_k(\mathbf{u}_k + L_k\mathbf{e}_k)$, (13)

Consider the problem (QILC) with the new cost function defined by:

$$J'_{k+1} = \mathbf{e}_{k+1}^T W_e \mathbf{e}_{k+1} + \mathbf{u}_{k+1}^T W_u \mathbf{u}_{k+1} + (\mathbf{u}_{k+1} - \hat{\mathbf{u}}_{k+1})^T (\mathbf{u}_{k+1} - \hat{\mathbf{u}}_{k+1}) \quad (14)$$

As before, to obtain the control update law that minimizes the cost function J_{k+1}', we get the following recursions

$$\boxed{\left.\begin{aligned} Q_{k+1} &= (I + W_u + T_u^T W_e T_u)^{-1}(Q_k + T_u^T W_e T_u) \\ Q_{k+1}L_{k+1} &= (I + W_u + T_u^T W_e T_u)^{-1}(Q_k L_k + T_u^T W_e) \\ \mathbf{u}_{k+1} &= Q_{k+1}(\mathbf{u}_k + L_{k+1}\mathbf{e}_k) \end{aligned}\right\} \quad (15)}$$

Remark (B1): From the recursion equations involving Q_{k+1} and L_{k+1}, it is clear that they are time varying and computationally intensive to implement. However, we can pre-compute these matrices offline and store them before an experiment is run.

***Lemma IV.B.1*:** The closed loop system using the modified Q-ILC controller designed above is stable if Q_0= I, L_0= 0.

Proof: The time varying system is stable if

$$\rho(Q_i(I - L_i T_u)) < 1 \,\forall i \quad (16)$$

Using eq. (15),

$$\begin{aligned} \rho(Q_i(I - L_i T_u)) &= \rho((I + W_u + T_u W_e T_u)^{-i} Q_0(I - L_0 T_u)) \\ &\le \beta^i, \quad \textit{if } Q_0 = I, L_0 = 0; \end{aligned}$$

where $\quad \beta = \sigma((I + W_e + T_u W_e T_u)^{-1}) < 1$ (17)

Therefore, $\rho(Q_i(I - L_i T_u)) < 1 \,\forall i$, and the closed loop time varying system is stable. ◊

The recursions are convergent if $\rho((I + W_e + T_u W_e T_u)^{-1}) < 1$. The steady state matrices are

$$\boxed{\begin{aligned} Q_{ss} &= (W_u + T_u^T W_e T_u)^{-1} T_u^T W_e T_u \\ Q_{ss}L_{ss} &= (W_u + T_u^T W_e T_u)^{-1} T_u^T W_e \end{aligned} \quad (18)}$$

Remark (B2): The steady state matrices Q_{ss} and L_{ss} are in fact, the solution to the Q-ILC problem using the cost function $J_{k+1} = \mathbf{e}_{k+1}^T W_e \mathbf{e}_{k+1} + \mathbf{u}_{k+1}^T W_u \mathbf{u}_{k+1}$ (19)

that is, a cost function without any penalty on the rate of change of the control input from one cycle to another. It can also be shown that assuming that there is no model uncertainty, this controller converges to the steady state in just one cycle.

Lemma IV.B.2: The convergence rate of the control input the modified Q-ILC scheme is given by:

$$\|\mathbf{u}_k - \mathbf{u}_\infty\|_2 \le \left(\prod_{i=1}^{k} \sigma(Q_i(I - L_i T_u))\right)\|\mathbf{u}_0 - \mathbf{u}_\infty\|_2 \qquad (20)$$
$$= \beta^{k(k+1)/2}\|\mathbf{u}_0 - \mathbf{u}_\infty\|_2$$

Remark (B3): Eq. (20) shows that the rate of convergence of the control input to the steady state value is much faster than the exponential linear convergence obtained by using standard QILC discussed in section IV A, since, the error converges monotonically as β^{k^2} -- *superlinear convergence*.

Though the rate of convergence is increased by the modified Q-ILC scheme, the performance of the controller must also be evaluated in terms of the steady state tracking error $\mathbf{e}_\infty$.

Lemma IV.B.3: The final steady state error $\mathbf{e}_\infty$ is given by the expression

$$\mathbf{e}_\infty = (I + T_u W_u^{-1} T_u^T W_e)^{-1}\mathbf{r} \qquad (11')$$

Remark (B4): While the rate of convergence for the proposed modified Q-ILC is much faster, the steady state output tracking error is the same as that obtained from the standard QILC implementation! At first glance, it seems as if the proposed controller is more 'effective' as compared to the standard QILC scheme. However, so far we have not compared the robustness of the two controllers.

V. Robustness Analysis of the Q-ILC Schemes

In practical applications, the model of the plant T_u is not known exactly. So, it is important to examine performance of any control system assuming that the true plant and the nominal plant (or plant model) are different. The robustness of the controller to modeling error has to be considered before implementation. In this analysis, we will consider only additive plant uncertainty, i.e., $T_u = \hat{T}_u + \Delta$, where $\hat{T}_u$ is the nominal plant, Δ is the model uncertainty. Such an uncertainty model also absorbs the case when we have nonzero initial conditions at the start of each cycle. We first design the learning controller based on the nominal model, and then determine the worst-case performance.

A. Robustness Analysis of "Standard" Q-ILC

The stability criteria is satisfied if $\rho(Q(I - LT_u)) < 1$. Using the nominal plant model in the controller,

$$\rho(Q(I - LT_u)) = \rho((I + W_u + \hat{T}_u^T W_e \hat{T}_u)^{-1}(I - \hat{T}_u^T W_e \Delta))$$
$$= \hat{\beta} > \beta$$

We see that the rate of convergence slows down if there is model uncertainty in the plant. Further, if the uncertainty is large, the closed loop system might even become unstable. Another important observation is that the larger the penalty on the output tracking error, W_e, the larger the effect of the plant uncertainty on stability and convergence. The maximum change in the rate of convergence of the learning controller is

$$\partial\beta = \hat{\beta} - \beta \le \frac{\overline{\sigma}(\hat{T}_u)\overline{\sigma}(W_e)\|\Delta\|_2}{\underline{\sigma}(I + W_u + \hat{T}_u^T W_e \hat{T}_u)} \qquad (21)$$

B. Robustness Analysis of the "Modified" Q-ILC

Since the closed loop system is time varying in this case, for stability, we need $\rho(Q_i(I - L_i T_u)) < 1\ \forall i$. As in section V.A., we use the nominal plant model in the controller,

$$Q_{k+1}(I - L_{k+1}T_u) = (I + W_u + \hat{T}_u^T W_e \hat{T}_u)^{-1}(Q_k(I - L_k T_u) + \hat{T}_u^T W_e \Delta)$$

First we will consider the case when we let $k \to \infty$. The matrices Q_{ss} and L_{ss} then determine the robustness of the learning algorithm. From eq. (18), using the nominal model in the controller,

$$\partial\beta = \hat{\beta} - \beta \le \frac{\overline{\sigma}(\hat{T}_u)\overline{\sigma}(W_e)\|\Delta\|_2}{\underline{\sigma}(I + W_u + \hat{T}_u^T W_e \hat{T}_u) - 1} \qquad (22)$$

Comparing eq. (21) and (22), we infer that the standard Q-ILC problem is more robust than the modified Q-ILC scheme. Further, it is natural to argue that instead of implementing time varying Q_k and L_k, we can just plug in the steady state solutions Q_{ss} and L_{ss}. However, the robustness of this scheme is extremely poor.

VI. An Alternative Iterative Learning Controller

From the discussion in sections IV and V, it is clear that there is a three-way tradeoff between robustness, performance and convergence (or learning) rate. In this section, we propose a controller based on the modified Q-ILC scheme, which cleverly uses the super-linear convergence and is robust at the same time. One of the reasons for the poor robustness of the modified Q-ILC scheme is the fact that the learning filters are cautious in the beginning and get more aggressive as iterations are completed. Therefore, the steady state closed loop system is not robust to modeling error. We wish to have a learning controller that is aggressive in the beginning and cautious as it approaches steady state. Hence, we start from the solution of the modified Q-ILC scheme at the time 'M', and then evolve the matrices backwards in time, from iteration to iteration as:

$$\boxed{\left.\begin{array}{l}\overline{Q}_k = Q_{M-k}\\ \overline{L}_k = L_{M-k}\end{array}\right\}\ \text{if } k < M \ \text{ and } \ \left.\begin{array}{l}\overline{Q} = I\\ \overline{L} = 0\end{array}\right\}\ \text{if } k > M \qquad (23)}$$

where, Q_{M-k} , L_{M-k} are the solutions to the modified Q-ILC problem at the $(M - k)^{\text{th}}$ iteration. By using the Q-ILC solution backwards in time we ensure that we are aggressive in the beginning and slowly become cautious. Eventually, the learning is turned off (after M iterations). Secondly, the super linear convergence rate is preserved, because we merely change the *order* in which the matrices are updated. Therefore, the control input error converges super-linearly for the first M iterations. Further, since the steady state values of the learning matrices are fixed at I and 0 respectively, we also have good robustness. By the above choice of the learning law, we can assure fast convergence (super-linear convergence attained by the modified Q-ILC), while ensuring robustness. The rate of convergence in this case is $\beta^{k(2M-(k+1))/2} >> \beta^{k(k+1)/2}$. Therefore, this scheme has much faster convergence initially than the modified Q-ILC scheme, and at $k = M$, the convergence rate is the same. In future, we will refer to this scheme as the backward Q-ILC scheme.

VII. RESULTS

For evaluating performance of the controllers discussed in sections IV and VI, we chose a model of a continuous time LTI system shown below:

$$G(s) = \left(\frac{8s^2 + 4s + 1}{s^3 + 3s^2 + 3s + 1} \right) \tag{24}$$

The reference trajectory was defined as shown in Fig. (2). The period of the trajectory was 0.4 seconds, and the sampling time was 0.008 seconds (N = 50).

In the first set of simulations, the proposed modified Q-ILC scheme was implemented on the same plant and reference trajectory as before. Figure (3) shows the plot of the tracking error when the steady state matrices Q_{ss} and L_{ss} given by eq. (18) are used in the control update law. The steady state error converges in just one cycle, with the final steady state error dependent on the ratio $(W_{\underline{u}})^{-1}W_e$. In fig. (4) the performance of the controller under a small model perturbation is plotted. It is observed that this scheme has very poor robustness.

The second set of simulations tests the performance, robustness and convergence of the backward Q-ILC scheme proposed in section VI. The design parameter M was chosen to be 8 in this case. For smaller steady state error, we can choose larger M. Figure (5) plots the tracking error for different values of the ratio $(W_{\underline{u}})^{-1}W_e$ assuming that there is no modeling uncertainty. As in the standard Q-ILC case, the steady state tracking error decreases on increasing the ratio. Figure (6) shows a plot of the robustness of the controller towards modeling uncertainty. The closed loop system is stable even for large uncertainties. Figure (7) shows a comparison of the nominal performance of the three proposed controllers under the same weights W_e and W_u. It is clear that under similar operating conditions, the rate of convergence of the backward Q-ILC scheme is much faster than the modified Q-ILC and the standard Q-ILC. The backward QILC scheme and the modified Q-ILC scheme have similar performance, and converge to almost the same error. The standard Q-ILC scheme has a much slower rate of convergence. Figure (8) shows that the backward Q-ILC has very good robustness and has faster convergence.

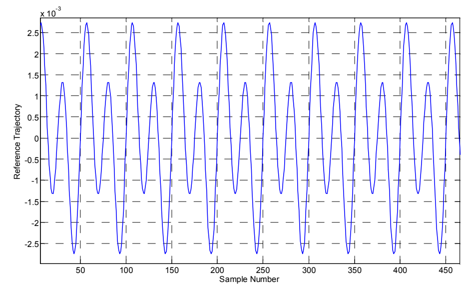

Fig. 2. Reference Trajectory for simulations

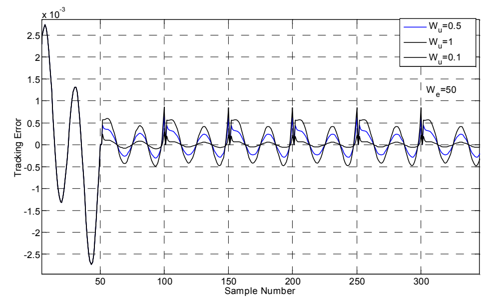

Fig. 3. Plot of tracking error when the steady state matrices $Q_{ss}$and L_{ss} are used in the ILC. Note that the convergence is in just one cycle for all choices of W_u. The steady state error is different.

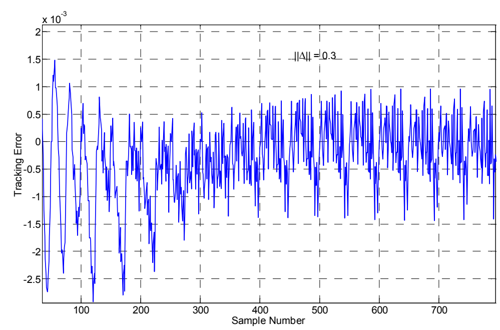

Fig. 4. Plot of tracking error (using the steady state solution to the modified Q-ILC problem) with modeling uncertainty $\|\Delta\|$=0.3

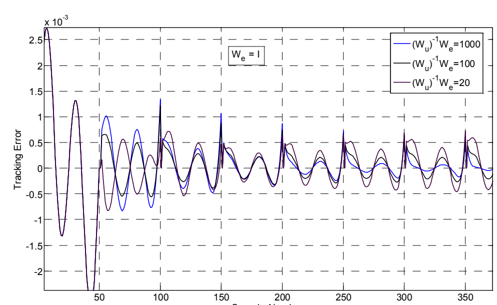

Fig. 5. Plot of tracking error for various choices of $(W_u)^{-1}W_e$ in the backward Q-ILC scheme. . Note the aggressive early convergence, and the cautious convergence after a few cycles.

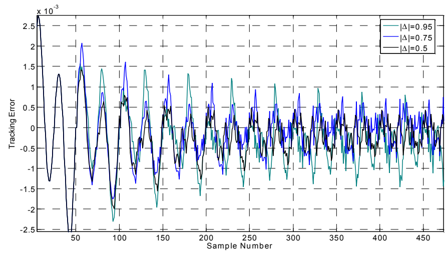

Fig. 6. Robustness of the backward Q-ILC scheme for various magnitudes of modeling uncertainty in the plant. ($W_u = 0.3$)..

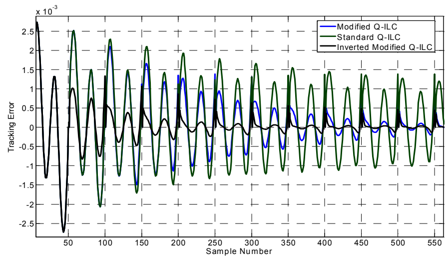

Fig. 7. Comparison of performance of the three controllers presented in this paper under nominal conditions (i.e., when the plant model is known exactly).

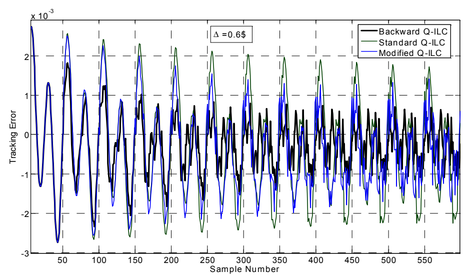

Fig. 8. Comparison of the robustness of the three controllers discussed in the paper under similar modeling errors, same choices of the weighting matrices W_e and W_u.

VIII. Summary and Conclusion

In this paper, three design techniques based on optimization of quadratic cost indices for iterative learning controllers were presented. Each design technique was developed in detail, and the effects of the design parameters on the closed loop system were studied. We also specified three criteria for evaluating the effectiveness of the iterative learning controller – steady state tracking error (performance), convergence rate (or rate of learning) and robustness. On closer analysis of the controllers based on theoretical results, it was predicted that there is a three-way tradeoff between these three metrics. The controllers were numerically simulated on an example plant. Secondly, a new controller design strategy was suggested that used a modified Q-ILC scheme to generate a learning law that is aggressive in the beginning and settles into a conservative scheme after the initial learning process is over. This gives us better convergence properties initially, and then robustness in the steady state. The two-fold advantage is that we achieve both good robustness and very high (super-linear) rates of convergence. Simulation results agree with predicted behavior.

References

[1] M. Uchiyama, "Formulation of high-speed motion pattern of a mechanical arm by trial"(in Japanese), *Trans. Soc. Instrum. Contr. Eng.,* vol.14, no.6, 1978, pp. 706-712.

[2] S. Arimoto, S. Kawamura and F.Miyazaki, "Bettering operation of robots by learning", *Jour. Robot. Syst.*, vol. 1, no. 2, 1984, pp. 123-140.

[3] G. Casalino and G. Bartolini, "A learning procedure for the control of movements of robotic manipulators", *Proc. IASTED Symp of Robot. and Auto.* , San Francisco, CA May 23-24, 1984, pp. 108-111.

[4] J. J. Craig, "Adaptive control of manipulators through repeated trials", *Proc. ACC*, San Diego, CA, June 1984, pp. 1566-1572.

[5] M. Norrlof and S. Gunnarsson, "Experimental comparison of some classical iterative control algorithms", *IEEE Trans. Robot. And Auto.*, vol. 18, no. 4, August 2002, pp. 636-641.

[6] N. Amann, D.H. Owens and E. Rogers, "Iterative learning control using prediction with arbitrary horizon", *Proc. Eur. Cont. Conf.* Brussels, Belgium, 1997.

[7] N. Amann, D.H. Owens and E. Rogers, "Predictive optimal iterative learning control", *Int. Jour. Control,* vol. 69, no. 2, 1998, 203-226.

[8] N. Amann, D.H. Owens and E. Rogers, "Iterative learning control for discrete-time systems with exponential rate of convergence", *IEEE Proc. Cont. Theory Appl.,* vol. 143, no. 2, 1996, 217-224.

[9] B. G. Dijkstra and O. H. Bogsra, "Extrapolation of optimal lifted system ILC solution, with application to a waferstage", *American controls Conference*, 2002.

[10] J. A. Frueh and M. Phan, "Linear quadratic optimal control (LQL)", *Int. Jour. Control,* vol. 73, no. 10, 2000, pp. 832-839.

[11] K. Feng and D. H. Owens, "Parameter optimization in iterative learning control", *Int. J. Control*, vol. 76, no. 11, 2003, pp. 1059-1069.

[12] R. Horowitz, "Learning control of robot manipulators", *J. Dynam. Syst. Meas. Contr.*, col. 115. pp. 402-411 .

[13] H. S. Lee and Z. Bien, "Robustness and Convergence of a PD-Type Iterative Learning Control"*, Iterative Learning Control: Analysis, Design, Integration and Applications.* Boston, Kluwer Acad., 1998.

[14] D. de Roover, "Synthesis of a robust iterative learning controller using an H_∞ approach", *Proc. 35th Conf. Decision and Cont.* , Kobe, Japan, pp. 3044-3049, 1996.

[15] M. Norloff, "An adaptive iterative learning algorithm with experiments on an industrial robot", *IEEE Trans. Robot. Auto.*, vol. 18, pp. 245-251, 2002.

[16] K. Furuta and M. Yamakita, "The design of a learning control system for multivariable systems", *Proc. IEEE Int. Symp. Intel. Cont.*, Philadelphia, Pennsylvania, USA, pp. 371-376, 1987.

[17] K. L. Moore, *Iterative learning control for deterministic systems,* Advances in Industrial Control Series, Springer-Verlag, London, UK, 1993.

[18] Z. Bien and J. –X. Xu, *Iterative Learning Control: Analysis, Design, Integration and Applications.* Norwell, MA, Kluwer Academic, 1998.

[19] D. de Roover, *Motion control of a wafer stage. A design approach for speeding up IC production,* PhD thesis, Delft University of Technology, 1997.

[20] M. A. Phan and R. W. Longman, "A mathematical theory of learning control for linear discrete multivariable systems", *Proc. of the AIAA/AAS Astro. Specialist Conference,* Minneapolis, MN, USA, pp. 740-746, 1988.

[21] R. Tousain, E. van der Meche and Okko Bosgra, "Designs of iterative learning control based on optimal control", *Proc. 40th IEEE Conf. Decis. Contr.*, Orlando, FL, 2001, pp. 4463-4468 vol.5.