

Analysis and Causal Formulation Proof of an Optimal Iterative Learning Algorithm

Vassiliki Vita¹, Marina Vassilaki¹, Panagiotis Karampelas^{1,2}, George E. Chatzarakis¹

¹Department of Electrical Engineering Educators,

School of Pedagogical and Technological Education (ASPETE), Athens, Greece

²IT Faculty, Hellenic American University, Athens, Greece

E-mail: vasvita@yahoo.co.uk, pkarampelas@hau.gr, geaxatz@otenet.gr

Received August 12, 2010; revised September 6, 2010; accepted September 19, 2010

Abstract

Iterative learning control (ILC) is used to control systems that operate in a repetitive mode, improving tracking accuracy of the control by transferring data from one repetition of a task, to the next. In this paper an optimal iterative learning algorithm for discrete linear systems is analyzed and a solution for its attainment is proposed. Finally the mathematical proof of the algorithm's causal formulation is also provided in its complete form, since its implementation requires its causal formulation.

Keywords: Casual Formulation, Discrete Systems, Iterative Learning Control

1. Introduction

Iterative Learning Control (ILC) is a relatively new concept in the control theory. It evolved from the need to control dynamical systems that are supposed to carry out a given task repetitively. More specifically for systems where the desired output values are a function of time and the task is carried out repeatedly. A classical example is the robot in the car industry that follows a given trajectory and welds at specific points along the trajectory. Normally the robot would be tuned once through feedback or feedforward control or even a combination of both. After the tuning, it would carry out the repetitions performing in the same way. The obvious drawback of this approach is the fact that if there is an error between the measured trajectory and the reference trajectory due to a wrong selection of the control input trajectory, the robot will repeat this error at each trial, *i.e.*, if there is an error in the performance it will repeat the same error at each iteration.

To overcome this problem, Arimoto, one of the inventors of ILC [1,2], suggested that both the information from the previous tasks or "trials" and the current task should be used to improve the control action during the current trial. In other words, the controller should learn iteratively the correct control actions in order to minimize the difference between the output of the system and the given reference signal. He called this method "bet-

terment process" [3]. This approach is more or less an imitation of the learning process of every intelligent being. Intelligent beings tend to learn by performing a trial (*i.e.* selecting a control input) and observing what was the end result of this control input selection. After that they try to change their behaviour (*i.e.* to pick up a new control input) in order to get an improved performance during the next trial. Because, a) the overall idea of ILC and the procedure results in a controller that learns the correct control input, and b) learning is done through iteration, the term Iterative Learning Control is nowadays used to describe control algorithms that result in the "betterment process" as suggested by Arimoto [3]. In this work an optimal iterative learning algorithm for discrete linear systems is analyzed and a solution for its attainment is proposed. Finally the mathematical proof of the algorithm's causal formulation is also provided in its complete form, since its implementation requires its causal formulation.

2. The Iterative Learning Control Aim

The aim of the ILC algorithm [4,5], is to find iteratively an optimal input u^* for the plant under investigation. This input, when applied to the plant, should generate an output y^* that tracks the desired output y_d "as accurately as possible". The use of the phrase "as accurately as possible" states the significance of obtaining the

smallest possible difference between the actual y^* and desired y_d output. This difference is actually the error e and the above lead to the conclusion that the error e is desired to be minimal. The ultimate aim of the ILC is to push the error e to zero. Thus it is clear that the applying an ILC algorithm will result in a two-dimensional system because the information propagates both along the time axis t and the iteration axis k .

The 2-dimensionality of the iterative learning control introduces problems in the analysis as well as the design of the system and hence the 2-dimensions are an expression of the systems dependent dynamics: a) the trial index k and b) the elapsed time t during each trial. In order to overcome such difficulties an algorithm for Iterative Learning Control is introduced, which has the property of solving the 2-dimensionality problem.

3. Proposed Cost Function

One way of solving this problem is to introduce the cost function for the system under investigation. The minimization of the cost function will provide an effective algorithm for iterative learning control (ILC).

The following cost function is proposed [4]:

$$J(u_{k+1}) = \|u_{k+1} - u_k\|_R^2 + \|e_{k+1}\|_Q^2 \quad (1)$$

where:

$J(u_{k+1})$ is the cost function with respect to the current trial input,
 $\|u_{k+1} - u_k\|_R^2$ is the norm of the difference between the current and previous trial inputs,
 $\|e_{k+1}\|_Q^2$ is the norm of the current error and
 R, Q are symmetric and positive definite weighing matrices. It is assumed that R and Q are diagonal matrices, and for simplicity $R = rI$ and $Q = qI$, where q and r are positive real numbers ($q, r \in R$).

In ILC literature, researchers have suggested different cost functions to solve the ILC problem. The reasons that lead to the realization that the selected cost function (1) is effective and appropriate are the following:

1) The $\|u_{k+1} - u_k\|_R^2$ factor represents the importance of keeping the deviation of the input between the trials small. Intuitively this should result in smooth convergence behavior. This requirement could also be stated as the need for producing smooth control signals, in order to obtain smooth manipulation of actuators.

2) The $\|e_{k+1}\|_Q^2$ factor represents the main objective of reducing the tracking error, at each iteration.

3) In order to state which of the above two factors plays a more significant role in the cost function the weighting matrices R and Q are used. If the interest is focused in retaining the deviation of the input between

the trials small, then the ratio $\beta = r/q$ has to be "large". On the other hand, if keeping the error small is more significant, then the ratio $\beta = r/q$ has to be "small". The actual meaning of "small" and "large" depends on the system being considered and the units measured.

4) The optimal value of the cost function is bounded. If the cost function is evaluated with $u_{k+1} = u_k$ then (1) becomes: $J(u_k) = \|u_k - u_k\|_R^2 + \|e_k\|_Q^2 = \|e_k\|_Q^2$ and hence the optimal value: $J(u_{k+1}) \leq \|e_k\|_Q^2$. It is also clear that the optimal cost function has a lower bound:

$J(u_{k+1}) = \|u_{k+1} - u_k\|_R^2 + \|e_{k+1}\|_Q^2 \geq \|e_{k+1}\|_Q^2$. Hence combining the above, the upper and lower bounds are expressed with (2).

$$\|e_{k+1}\|_Q^2 \leq J_{k+1}(u_{k+1}) \leq \|e_k\|_Q^2 \quad (2)$$

4. The Norm-Optimal ILC Algorithm

The differentiation of the cost function (1) with respect to u_{k+1} produces the solution used to update the input, the norm-optimal ILC algorithm, which is investigated in this work. The input up-date law is the following [4]:

$$\frac{\partial J}{\partial u_{k+1}} = 0 \Rightarrow$$

$$u_{k+1} = u_k + R^{-1}G^T Q e_{k+1} = u_k + G^* e_{k+1} \quad (3)$$

or

$$u_{k+1} - u_k = R^{-1}G^T Q e_{k+1} \quad (4)$$

where:

u_{k+1} is the current trial input,
 u_k is the previous trial input,
 e_{k+1} is the current trial error and
 $G^* = R^{-1}G^T Q$ is the gain matrix-adjoint of the plant, that represents the relative weighting of the cost function requirements (error-input deviation).

5. Characteristics and Properties of the ILC Algorithm

The main characteristics of the update law for the proposed Iterative learning control scheme (3) can be summarized to the following:

1) Non-causality. The dependence of the input update law (3) on the adjoint of the plant G^* , where G^* is a function of the error in future sample times than the current one, implies non-causality.

2) Integrator-type of approach. The algorithm utilizes the input change instead of the input itself to evaluate the performance of the system.

3) Feedback approach. This characteristic is one of the

most important ones. Several authors [2] suggested feedforward approaches where the error of the previous trial was used in the input update law: $u_{k+1} = u_k + Ke_k$. On the contrary the update law in (3) is of the form: $u_{k+1} = u_k + Ke_{k+1}$, which means that it uses feedback of the current trial error. The feedback operator K is in this case the adjoint of the plant G^* . The feedforward approaches face the problem of ignoring the current trial effects such as plant modeling errors and possible disturbances. This means that the algorithm is not sensitive to these effects. On the contrary the feedback approach of (3) takes into account all of this producing in this way a more robust solution and this is likely to be a more effective one in the plant stabilizing sense.

4) Descent algorithm. With the help of the equation: $\|e_{k+1}\|^2 \leq J_{k+1} \leq \|e_k\|^2$, it is deduced that the norm of the error $\|e_{k+1}\|$ is monotonically decreasing from trial to trial.

Two of the most important properties of the algorithm are presented below:

1) Convergence of the error sequence to zero. It can be shown not only that the error e converges to a final value e_∞ as the number of iterations $k \rightarrow \infty$ but also that $e_\infty = 0$. It is significant to mention that the convergence is exponential. It is also significant to be mentioned at this point, that the error sequence convergence can be proved even for an arbitrary chosen initial error e_0 . This is a form of robustness, since it is ensured that the error will converge to zero, independent of the initial error value.

2) Convergence of the input sequence. It can be shown that the input of the algorithm converges to a final value u_∞ as the number of trials $k \rightarrow \infty$.

6. Causal Formulation of the ILC Algorithm

One of the main characteristics of the ILC algorithm (3) is that the algorithm seems to be non-causal. Because of its characteristics it is a very efficient algorithm in terms of plant control. The main drawback is the fact that enormous amount of data and computational effort is required due to the matrices size. A solution to this problem would be the implementation of the algorithm. On the other hand, in order to be able to implement the algorithm it is essential to derive it in a causal form.

6.1. The Proposed Causal Algorithm

The causal solution to the above problem is given by introducing the following proposed algorithm for Norm-Optimal ILC, which consists of the following terms:

Term 1: The gain matrix $K(t)$. Given in the form of the discrete Ricatti equation [6,7]:

$$K(t) = \Phi^T K(t+1) \Phi - \Phi^T K(t+1) \Gamma \left[\Gamma^T K(t+1) \Gamma + R \right]^{-1} \Gamma^T K(t+1) \Phi + C^T Q C \quad (5)$$

for $t \in [0, N-1]$ and with the terminal condition $K(N) = 0$.

Term 2: The feedforward (predictive) term.

$$\xi_{k+1}(t) = \left[I + K(t) \Gamma R^{-1} \Gamma^T \right]^{-1} \left\{ \Phi^T \xi_{k+1}(t+1) + C^T Q e_k(t+1) \right\} \quad (6)$$

for $t \in [0, N-1]$ with the terminal condition $\xi_{k+1}(N) = 0$.

Term 3: The input update law.

$$u_{k+1}(t) = u_k(t) - \left(\Gamma^T K(t) \Gamma + R \right)^{-1} \Gamma^T K(t) \Phi \left[x_{k+1}(t) - x_k(t) \right] + R^{-1} \Gamma^T \xi_{k+1}(t) \quad (7)$$

Easily can be observed that Term I is independent of the inputs, outputs and states of the system, Term II, the predictive term $\xi_{k+1}(t)$ is dependent on the previous trial error $e_k(t+1)$ and Term III, the input update law depends on the previous trial input $u_k(t)$, the current state $x_{k+1}(t)$, the previous trial state $x_k(t)$, and the predictive term $\xi_{k+1}(t)$. This is hence a causal iterative learning control algorithm consisting of current trial-full state-feedback along with feedforward of the previous trial error data.

6.2. Mathematical Proof of the Causal Formulation

In this section a complete mathematical proof of the algorithm is provided, demonstrating all the necessary steps from the optimization problem to the conception, development and finally the acquisition of the three terms of the algorithm's causal form. The proof is based on the general idea proposed in [8] but it is for the first time that it is in its complete form, as it includes all the intermediate steps and calculations in an analytic way.

The proof is as follows: Rewriting the input update law as a difference between the current trial input and the previous trial input gives the relationship (4): $u_{k+1}(t) - u_k(t) = R^{-1} G^T Q e_{k+1}$. This form of the input update law, when written in super-vector notation results in relationship (8).

$$\begin{bmatrix} u_{k+1}(0) - u_k(0) \\ u_{k+1}(1) - u_k(1) \\ u_{k+1}(2) - u_k(2) \\ \dots \\ u_{k+1}(N-1) - u_k(N-1) \end{bmatrix} = R^{-1} \begin{bmatrix} \Gamma^T C^T & \Gamma^T \Phi^T C^T & \Gamma^T (\Phi^T)^2 C^T & \dots & \Gamma^T (\Phi^T)^{N-1} C^T \\ 0 & \Gamma^T C^T & \Gamma^T \Phi^T C^T & \dots & \Gamma^T (\Phi^T)^{N-2} C^T \\ 0 & 0 & \Gamma^T C^T & \dots & \Gamma^T (\Phi^T)^{N-3} C^T \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \Gamma^T C^T \end{bmatrix} Q \begin{bmatrix} e_{k+1}(1) \\ e_{k+1}(2) \\ e_{k+1}(3) \\ \dots \\ e_{k+1}(N) \end{bmatrix} \quad (8)$$

The difference of the control input

$u(t) = u_{k+1}(t) - u_k(t)$ at the t_{th} sample (by examining the $(t+1)^{th}$ row of (8)) with $0 \leq t \leq N$, can be written:

$$u_{k+1}(t) - u_k(t) = \sum_{j=t+1}^N R^{-1} \Gamma^T (\Phi^T)^{j-t-1} C^T Q e(j) \quad (9)$$

$$\Leftrightarrow u_{k+1}(t) - u_k(t) = R^{-1} \Gamma^T \sum_{j=t+1}^N (\Phi^T)^{j-t-1} C^T Q e(j) \quad (10)$$

In order to derive a state space description of the adjoint system, the discrete co-state $p(t)$ is introduced, such that:

$$u_{k+1}(t) - u_k(t) = R^{-1} \Gamma^T p(t) \quad (11)$$

Comparing (10) and (11) it follows that the co-state satisfies:

$$p(t) = \sum_{j=t+1}^N (\Phi^T)^{j-t-1} C^T Q e(j) \quad 0 \leq t \leq N \quad (12)$$

By expanding the sums in (12):

$$\begin{aligned} p(t) &= \sum_{j=t+1}^N (\Phi^T)^{j-t-1} C^T Q e(j) \Rightarrow \\ p(t) &= (\Phi^T)^0 C^T Q e(t+1) + \Phi^T C^T Q e(t+2) + \\ & (\Phi^T)^2 C^T Q e(t+3) + \dots + (\Phi^T)^{N-t-1} C^T Q e(N) \\ &= C^T Q e(t+1) + \Phi^T C^T Q e(t+2) + \\ & (\Phi^T)^2 C^T Q e(t+3) + \dots + (\Phi^T)^{N-t-1} C^T Q e(N) \\ &= C^T Q e(t+1) + \Phi^T C^T Q e(t+2) + \\ & \Phi^T [C^T Q e(t+3) + \dots + (\Phi^T)^{N-t-2} C^T Q e(N)] \end{aligned} \quad (13)$$

but

$$\begin{aligned} p(t+1) &= C^T Q e(t+2) + \Phi^T C^T Q e(t+3) + \dots \\ & + (\Phi^T)^{N-t-2} C^T Q e(N) \end{aligned} \quad (14)$$

Hence the above yields the recursion relation for $p(t)$:

$$\Rightarrow p(t) = C^T Q e(t+1) + \Phi^T p(t+1) \quad 0 \leq t \leq N \quad (15)$$

with $p(N-1) = C^T Q e(N)$ or equivalently $p(N) = 0$.

Consequently, the complete discrete input update law is as follows:

$$u_{k+1}(t) = u_k(t) + R^{-1} \Gamma^T p_{k+1}(t), \quad 0 \leq t \leq N \quad (16)$$

$$\Rightarrow u_{k+1}(t) - u_k(t) = R^{-1} \Gamma^T p_{k+1}(t) \quad (17)$$

and

$$p_{k+1}(t) = C^T Q e_{k+1}(t+1) + \Phi^T p_{k+1}(t+1), \quad p_{k+1}(N) = 0 \quad (18)$$

The main observation is the fact that the adjoint system has a terminal condition (at $t = N$) instead of an initial condition; this is a sign of a non-causal solution.

Equation (18) is an expression for the current trial co-state. Another expression for the co-state $p_{k+1}(t)$ can be calculated as follows: A standard technique [6,7] is to set the discrete co-state equal to:

$$\begin{aligned} p_{k+1}(t) &= \\ & -K(t) (I + \Gamma R^{-1} \Gamma^T K(t))^{-1} \Phi [x_{k+1}(t) - x_k(t)] + \xi_{k+1}(t) \end{aligned} \quad (19)$$

with a gain matrix $K(t)$ and a correction term $\xi_{k+1}(t)$. Both the terms $K(t)$ and $\xi_{k+1}(t)$ should be computed before each trial. Lets set:

$$M(t) = -K(t) [I + \Gamma R^{-1} \Gamma^T K(t)]^{-1} \quad (20)$$

and then substitute this into equation (19):

$$\begin{aligned} \Rightarrow p_{k+1}(t) &= -K(t) (I + \Gamma R^{-1} \Gamma^T K(t))^{-1} \Phi [x_{k+1}(t) - x_k(t)] \\ & + \xi_{k+1}(t) \\ \Rightarrow p_{k+1}(t) &= M(t) \Phi [x_{k+1}(t) - x_k(t)] + \xi_{k+1}(t) \end{aligned} \quad (21)$$

Setting the functions $f(t)$ and $g(t)$ as follows easily can be observed that these are independent of the states.

$$\begin{aligned} f(t) &= M(t) + [I + K(t) \Gamma R^{-1} \Gamma^T]^{-1} C^T Q C - \\ & [I + K(t) \Gamma R^{-1} \Gamma^T]^{-1} \Phi^T M(t+1) \Phi \end{aligned}$$

and

$$\begin{aligned} g(t) &= [I + K(t) \Gamma R^{-1} \Gamma^T]^{-1} \Phi^T \xi_{k+1}(t+1) + \\ & [I + K(t) \Gamma R^{-1} \Gamma^T]^{-1} C^T Q e_k(t+1) - \xi_{k+1}(t) \end{aligned}$$

1) Calculating the (feedforward) predictive term $\xi_{k+1}(t+1)$

By setting $g(t) = 0$:

$$g(t) = [I + K(t)\Gamma R^{-1}\Gamma^T]^{-1} \Phi^T \xi_{k+1}(t+1) + \\ [I + K(t)\Gamma R^{-1}\Gamma^T]^{-1} C^T Q e_k(t+1) - \xi_{k+1}(t) = 0$$

Solving the above relationship for the predictive term $\xi_{k+1}(t+1)$ results in:

$$\xi_{k+1}(t) = [I + K(t)\Gamma R^{-1}\Gamma^T]^{-1} \{ \Phi^T \xi_{k+1}(t+1) + \\ C^T Q e_k(t+1) \} \quad (22)$$

which is simply the desired expression for the (feedforward) predictive term.

2) Calculating the gain matrix $K(t)$

By setting $f(t) = 0$:

$$f(t) = M(t) + [I + K(t)\Gamma R^{-1}\Gamma^T]^{-1} C^T Q C - \\ [I + K(t)\Gamma R^{-1}\Gamma^T]^{-1} \Phi^T M(t+1) \Phi = 0 \\ \Rightarrow M(t) + [I + K(t)\Gamma R^{-1}\Gamma^T]^{-1} C^T Q C - \\ [I + K(t)\Gamma R^{-1}\Gamma^T]^{-1} \Phi^T M(t+1) \Phi = 0$$

Multiplying both sides of the above relationship with $[I + K(t)\Gamma R^{-1}\Gamma^T]$ yields:

$$[I + K(t)\Gamma R^{-1}\Gamma^T] M(t) + C^T Q C \\ - \Phi^T M(t+1) \Phi = 0 \quad (23)$$

Substitute $M(t)$ with its equivalent from equation (20), equation (23) becomes:

$$-[I + K(t)\Gamma R^{-1}\Gamma^T] K(t) [I + \Gamma R^{-1}\Gamma^T K(t)]^{-1} + \\ C^T Q C + \Phi^T K(t+1) [I + \Gamma R^{-1}\Gamma^T K(t+1)]^{-1} \Phi = 0 \\ \Rightarrow -[I + K(t)\Gamma R^{-1}\Gamma^T] [I + K(t)\Gamma R^{-1}\Gamma^T]^{-1} K(t) + \\ C^T Q C + \Phi^T K(t+1) [I + \Gamma R^{-1}\Gamma^T K(t+1)]^{-1} \Phi = 0 \\ \Rightarrow -K(t) + C^T Q C + \\ \Phi^T K(t+1) [I + \Gamma R^{-1}\Gamma^T K(t+1)]^{-1} \Phi = 0$$

Solving the above term for the gain matrix $K(t)$ results in the following relationship:

$$K(t) = C^T Q C + \Phi^T K(t+1) [I + \Gamma R^{-1}\Gamma^T K(t+1)]^{-1} \Phi \\ \Rightarrow K(t) = C^T Q C + \Phi^T K(t+1) \\ \left\{ I - \Gamma R^{-1}\Gamma^T K(t+1) [I + \Gamma R^{-1}\Gamma^T K(t+1)]^{-1} \right\} \Phi$$

further manipulations and expanding the brackets of the above equation result in the desired relationship of the gain matrix (5) which is the usual discrete Matrix Riccati equation:

$$K(t) = \Phi^T K(t+1) \Phi -$$

$$\Phi^T K(t+1) \Gamma [\Gamma^T K(t+1) \Gamma + R]^{-1} \Gamma^T K(t+1) \Phi + C^T Q C$$

3) Calculating the input update law for the new input $u_{k+1}(t)$

In order to prove the desired relationship for the new input, equation (16) is used, which is:

$$\Rightarrow u_{k+1}(t) = R^{-1}\Gamma^T p_{k+1}(t) + u_k(t)$$

Substituting in the above equation for the discrete co-state $p_{k+1}(t)$ with its equivalent from equation (18) the following equation is produced:

$$u_{k+1}(t) = R^{-1}\Gamma^T \{ -K(t) (I + \Gamma R^{-1}\Gamma^T K(t))^{-1} \Phi [x_{k+1}(t) \\ - x_k(t)] + \xi_{k+1}(t) \} + u_k(t)$$

By expanding the brackets:

$$\Rightarrow u_{k+1}(t) = -R^{-1}\Gamma^T K(t) (I + \Gamma R^{-1}\Gamma^T K(t))^{-1} \Phi [x_{k+1}(t) \\ - x_k(t)] + R^{-1}\Gamma^T \xi_{k+1}(t) + u_k(t)$$

$$\Rightarrow u_{k+1}(t) = -(I + R^{-1}\Gamma^T K(t)\Gamma)^{-1} R^{-1}\Gamma^T K(t) \Phi [x_{k+1}(t) \\ - x_k(t)] + R^{-1}\Gamma^T \xi_{k+1}(t) + u_k(t)$$

Having in mind that the multiplication at any desired point within the equation with the identity matrix I there is no loss of the equality:

$$\Leftrightarrow u_{k+1}(t) = -(I + R^{-1}\Gamma^T K(t)\Gamma)^{-1} * I * R^{-1}\Gamma^T K(t) \cdot$$

$$\Phi [x_{k+1}(t) - x_k(t)] + R^{-1}\Gamma^T \xi_{k+1}(t) + u_k(t)$$

$$\Rightarrow u_{k+1}(t) = -(I + R^{-1}\Gamma^T K(t)\Gamma)^{-1} * R^{-1}R * R^{-1}\Gamma^T \cdot$$

$$K(t) \Phi [x_{k+1}(t) - x_k(t)] + R^{-1}\Gamma^T \xi_{k+1}(t) + u_k(t)$$

$$\Rightarrow u_{k+1}(t) = -(R + RR^{-1}\Gamma^T K(t)\Gamma)^{-1} RR^{-1}\Gamma^T K(t) \cdot$$

$$\Phi [x_{k+1}(t) - x_k(t)] + R^{-1}\Gamma^T \xi_{k+1}(t) + u_k(t)$$

$$\Rightarrow u_{k+1}(t) = -(R + \Gamma^T K(t)\Gamma)^{-1} \Gamma^T K(t) \cdot$$

$$\Phi [x_{k+1}(t) - x_k(t)] + R^{-1}\Gamma^T \xi_{k+1}(t) + u_k(t)$$

After the above manipulations and simplifications the following relationship comes out, which is simply the expression for the causal Term III (7), the input up-date law:

$$u_{k+1}(t) = u_k(t) - (\Gamma^T K(t)\Gamma + R)^{-1} \Gamma^T K(t) \cdot$$

$$\Phi [x_{k+1}(t) - x_k(t)] + R^{-1}\Gamma^T \xi_{k+1}(t)$$

7. Conclusion

In this paper an optimal iterative learning algorithm for discrete systems is analyzed. The algorithm's properties and characteristics were presented with the most important feature to be that the error sequence converges to zero with an exponential rate for the case of discrete time plants. Finally, the complete causal formulation of the algorithm was derived in detail, consisting of optimal state feedback and optimal prediction feedforward.

8. References

- [1] Y. Chen and C. Wen, "Iterative Learning Control-Convergence, Robustness and Applications," Springer Verlag, London, 1999.
- [2] Z. Bien and J. X. Xu, "Iterative Learning Control-Analysis, Design, Integration and Applications," Kluwer Academic Publishers, Dordrecht, 1998.
- [3] S. Arimoto, S. Kawamura and F. Miyazaki, "Bettering Operation of Dynamic Systems by Learning: A New Control Theory for Servomechanism or Mechatronic Systems," *Proceedings 23rd IEEE Conference on Decision and Control*, Las Vegas, Nevada, 1984, pp. 1064-1069.
- [4] N. Amann, D. H. Owens and E. Roger, "Iterative Learning Control for Discrete Time Systems with Exponential Rate of Convergence," *IEE Proceeding of the Institution of Electrical Engineers on Control Theory and Applications*, Vol. 143, No. 2, 1996, pp. 217-224.
- [5] K. L. Moore, "Iterative Learning Control for Deterministic Systems," *Advances in Industrial Control Series*, Springer-Verlag, London, 1993.
- [6] K. J. Astrom and B. Wittenmark, "Computer controlled systems," 2nd Edition, Prentice Hall, Englewoods Cliffs, 1990.
- [7] T. Kailath, "Linear Systems," Prentice Hall, Englewoods Cliffs, 1980.
- [8] N. Amann, "Optimal Algorithms for Iterative Learning Control," Ph.D. Dissertation, University of Exeter, UK, 1996.