Experimentally Verified Robustness Properties of a Class of Model Inverse ILC Algorithms

Jari Hatonen[a], James D. Ratcliffe[b], Paul L. Lewin[b], Eric Rogers[b], David H. Owens[a]

[a]Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, S1 3JD, UK

[b] School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK

In this paper, the subject is the robustness and properties of an inverse type iterative learning control algorithm. In addition to new theoretical results experimental results from application to an industrial-scale gantry robot system are given.

1. Introduction

Iterative Learning Control (ILC) is a relatively new addition to the toolbox of control algorithms. ILC is concerned with the performance of systems that operate in a repetitive manner. Such systems include robot arm manipulators and chemical batch processes, where the task of following some specified output trajectory $r(t)$ in an interval $t \in [0, N]$ with high precision is repeated time and time again. The use of conventional control algorithms with such systems will result in the same level of tracking error being repeated time and time again. Motivated by human learning, the basic idea of ILC is to use information from previous executions of the task in order to improve performance from trial to trial in the sense that the tracking error is sequentially reduced from trial-to-trial. For further background on ILC see, as two representatives of the very large literature [1] and/or [2] and the references in this survey paper.

The concept of inverting plant dynamics to achieve perfect tracking is a simple and obvious one. However, it is hesitantly used in high precision tasks as uncertainty in plant models can lead to sub-optimal tracking and potential stability issues. Inverse models also tend to amplify measurement noise, which makes them even less attractive in feedback control applications.

This paper produces new results on how inverse models can be effectively used in the context of ILC. In particular, the robustness and noise rejection properties of an inverse model ILC algorithm are studied in both analysis and experiment. The experimental work is based on a gantry robot system and it is important to note that this is by no means the first application of ILC to robots see, for example, [3] — in fact, the robot system here is used to (begin the) experimental benchmarking of one algorithm and in essence this is where the focus of this paper lies.

2. Problem definition

As a starting point consider the following standard linear, time-invariant single input, single output state-space representation defined over *finite* time interval, $t \in [0, N]$ (in order to shorten notation it is assumed that sampling time is unity):

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \qquad x(0) = x_0 \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

where the state $x(\cdot) \in \mathbb{R}^n$, output $y(\cdot) \in \mathbb{R}$, input $u(\cdot) \in \mathbb{R}$ and $x(0) = 0$. From now on it will be assumed that $CB > 0$ and that the system (1) is controllable and observable. Furthermore, a reference signal $r(t)$ is specified and the control objective is to find an input function $u(t)$ so that the output function $y(t)$ tracks the reference signal $r(t)$ as accurately as possible.

Suppose now that the process is required to repeat the same operation over the finite time interval and is rest set before the start of each new iteration or trial. Then in essence ILC seeks to use the information generated on previous trials to iteratively (i.e. from trial-to-trial) learn the control signal required to produce the desired reference signal. One of the possible problems which can be formulated in this setting is the design of a control law of the form

$$u_{k+1} = f(u_k, u_{k-1}, \dots u_{k-r}, e_{k+1}, e_k, \dots, e_{k-s}) \tag{2}$$

such that

$$\lim_{k\to\infty} \|e_k\| = 0 \quad \lim_{k\to\infty} \|u_k - u^*\| = 0 \tag{3}$$

where $u_k = [u_k(0)\ u_k(1)\ \dots\ u_k(N)]^T$, $y_k = [y_k(0)\ y_k(1)\ \dots\ y_k(N)]^T$, $e_k = [r(0)-y_k(0)\ r(1)-y_k(1)\ \dots\ r(N)-y_k(N)]^T$, (i.e. the error on trial k) and $\|\cdot\|$ is a suitable norm. Note that if the mapping f in (2) is not a function of e_{k+1}, then it is typically said that the algorithm is of feedforward type, otherwise it is of feedback type.

For analysis purposes, note that because the system (1) is defined over a finite time-interval, it can be represented equivalently with a matrix equation $y_k = G_e u_k$, where

$$G_e = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \dots & \dots & 0 \end{bmatrix} \tag{4}$$

where the elements CA^jB of the matrix G_e are the Markov parameters of the plant (1). It is assumed here that the reference signal $r(t)$ satisfies $r(0) = Cx_0$. Then it can be shown (see [[4]]) that for analysis it is sufficient to consider a "lifted" plant equation $y_{k,l} = G_{e,l}u_{k,l}$ where $u_{k,l} = [u_k(0)\ u_k(1)\ \dots\ u_k(N-1)]^T$, $y_{k,l} = [y_k(1)\ y_k(2)\ \dots\ y_k(N)]^T$ and

$$G_{e,l} = \begin{bmatrix} CB & 0 & 0 & \dots & 0 \\ CAB & CB & 0 & \dots & 0 \\ CA^2B & CAB & CB & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \dots & \dots & CB \end{bmatrix} \tag{5}$$

Note that because it was assumed that $CB \neq 0$, $G_{e,l}$ is invertible, and consequently for an arbitrary reference r there exists u^* so that $r = G_{e,l}u^*$. Hence it would appear that this inverse model algorithm can be regarded as theoretically "perfect". This, however, would require an exact system model to be available and implemented which is not a practically justified assumption — the best is that a nominal model is available or chosen deliberately to reduce the computational burden. Here this 'lifted' plant will be used as a starting point for analysis, and in order to shorten notation, the subscript l will be omitted.

3. The Inverse Model algorithm

There are many possible inverse plant ILC algorithms and here as a representative we consider the case when

$$u_{k+1} = u_k + G_e^{-1} e_k \tag{6}$$

Simple analysis of the corresponding error evolution equation shows the expected result that error converges to zero in one iteration which is the perfect "solution". This requires the "perfect" model G_e and in practice it has to be replaced by a nominal model denoted here by G_o, i.e.

$$u_{k+1} = u_k + G_o^{-1} e_k \tag{7}$$

This yields the following error evolution equation:

$$e_{k+1} = (I - G_e G_o^{-1}) e_k \tag{8}$$

The convergence characteristics of (8) depend upon the matrix $G_e G_o^{-1}$, a matrix which has no guarantee of stability. A simple attempt to introduce stability is to insert a scalar gain, β, into the algorithm.

$$u_{k+1} = u_k + \beta G_o^{-1} e_k \tag{9}$$

and hence

$$e_{k+1} = (I - \beta G_e G_o^{-1}) e_k \tag{10}$$

A necessary and sufficient condition for stability is for the spectral radius of $(I - \beta G_e G_o^{-1})$ to be less than 1 but satisfying this may still lead to very poor performance of the algorithm. This paper allows β to vary in such a manner that the l_2-norm of the error is monotonically decreasing which is obviously a very useful property of an ILC algorithm. More precisely, the update equation and the error dynamics are determined by

$$u_{k+1} = u_k + \beta_{k+1} G_o^{-1} e_k \tag{11}$$

$$e_{k+1} = (I - \beta_{k+1} G_e G_o^{-1}) e_k \tag{12}$$

Norm Optimal Iterative Learning Control (NOILC) [5] is one optimal ILC routine that has been shown to give monotonic error convergence in spite of some model uncertainties. NOILC minimises both the error and the change in input between trials by computing $\min_{u_{k+1} \in \mathbb{R}^N} J(u_{k+1})$ where the cost function $J(u_{k+1})$ is given [7] by

$$J(u_{k+1}) = \|e_{k+1}\|^2 + \|u_{k+1} - u_k\|^2 \tag{13}$$

where the l_2-norm is used. This framework extends to the use of the adaptive update law (11) by using

$$J(\beta_{k+1}) = \|e_{k+1}\|^2 + w\beta_{k+1}^2 \tag{14}$$

where w can be freely chosen such that $w > 0$. This cost function adds flexibility whilst still maintaining the NOILC ideal of minimising error and smoothing changes in input. For the case $G_o = G_e$ a straightforward minimisation of (14) yields an optimal solution:

$$\beta_{k+1} = \frac{\|e_k\|^2}{w + \|e_k\|^2} \tag{15}$$

A convergence analysis of this algorithm for the case $G_o = G_e$ is given next.

Theorem 1 *If $G_o = G_e$, $w \in \mathbb{R}$, $w > 0$ then $\|e_{k+1}\| < \|e_k\|$ if $e_k \neq 0$. Furthermore,*

$$\lim_{k\to\infty} \|e_k\| = 0 \quad \text{and} \quad \lim_{k\to\infty} \beta_{k+1} = 0 \tag{16}$$

demonstrating monotonic convergence to zero tracking error.

Proof 1 *Selecting a sub-optimal choice $\beta_{k+1} = 0$ in the cost function (14) yields $J(0) = \|e_k\|^2$. Since this choice is sub-optimal it follows:*

$$\|e_k\|^2 \geq \|e_{k+1}\|^2 + w\beta_{k+1}^2 \geq \|e_{k+1}\|^2 \tag{17}$$

demonstrating monotonic convergence. Reformulation of (17) gives

$$\|e_k\|^2 - w\beta_{k+1}^2 \geq \|e_{k+1}\|^2 \geq 0 \tag{18}$$

and applying induction further gives

$$\|e_0\|^2 - w\sum_{i=1}^{k+1} \beta_i^2 \geq 0 \tag{19}$$

and because k is arbitrary, $\lim_{k\to\infty} \beta_k = 0$. *This results in*

$$0 = \lim_{k\to\infty} \beta_{k+1} = \lim_{k\to\infty} \frac{\|e_k\|^2}{w + \|e_k\|^2} \tag{20}$$

This is only possible if $\lim_{k\to\infty} e_k = 0$. *Furthermore, the interlacing result (17) implies that $\|e_{k+1}\| < \|e_k\|$ if $e_k \neq 0$ and the proof is complete.*

1. □

Note 1 *Note that the choice of $w = w_1\|e_k\|^2$ in (15) where $G_o = G_e$ yields the error evolution equation $e_{k+1} = (1 - (1 + w_1)^{-1})e_k$. Then for any $w_1 > 0$ error convergence is geometric.*

4. Robustness of the Inverse Model algorithm

Robustness of ILC algorithms is an active research topic and space limitations preclude a summary of the many approaches and their relative limitaitons/merits. Here we undertake an analysis of the algorithm of the previous section both in terms of system stability and performance by retaining monotonic convergence in the presence of model uncertainty. For this we need an uncertainty representation and here we consider the case when the true plant, $G_e \neq G_o$, and the model uncertainty of G_o is taken to be a multiplicative matrix U, i.e. $G_e = G_o U$. The first result is as follows.

Theorem 2 *Suppose $U + U^T$ is a positive-definite matrix. If $e_k \neq 0$ there exists a $\beta_{k+1} > 0$ such that $\|e_{k+1}\|^2 - \|e_k\|^2 < 0$. Furthermore the value of such β_{k+1} has to satisfy the following inequality*

$$v^T(\frac{1}{\beta_{k+1}}I - U)^T(\frac{1}{\beta_{k+1}}I - U)v < \frac{1}{\beta_{k+1}^2}\|v\|^2 \quad (21)$$

where $v \in \mathbb{R}^N$ is arbitrary.

Proof 2 *Use of (9) yields*

$$\|e_{k+1}\|^2 - \|e_k\|^2 = -2\beta_{k+1}e_k^TUe_k + \beta_{k+1}^2 e_k^TU^TUe_k < 0 \quad (22)$$

Since $U + U^T$ is assumed to be a positive-definite matrix and $\beta_{k+1} > 0$, the terms $-2\beta_{k+1}e_k^TUe_k$ and $\beta_{k+1}^2 e_k^TU^TUe_k$ are, for an arbitrary nonzero e_k, strictly positive and strictly negative respectively. Then for $\|e_{k+1}\|^2 < \|e_k\|^2$ it is necessary that the following inequality must be true.

$$2\beta_{k+1}e_k^TUe_k > \beta_{k+1}^2 e_k^TU^TUe_k \quad (23)$$

Since the left hand term of inequality (23) is of $O(\beta_{k+1})$ and the right hand term is of $O(\beta_{k+1}^2)$ it shows that the inequality is met for a sufficiently small β_{k+1}, giving monotonic convergence. Completing the square in (23) now gives (21).

Theorem 2 shows that if (21) holds true then error convergence is monotonic. The next proposition further shows that under this condition the error converges to zero. The proof of this result follows from that of Proposition 2 and hence the details are omitted here.

Theorem 3 *If the condition in Theorem 2 holds then $\lim_{k\to\infty} e_k = 0$.*

Note 2 *Note that it is easy to show that a sufficient condition for $U + U^T$ to be a positive-definite system is that the underlying system $U(z)$ corresponding to U is a positive-real system. The phase shift of such a system lies within $\pm 90^o$ for all frequencies. Therefore the algorithm can tolerate a plant uncertainty of $\pm 90^o$ phase shift for all frequencies.*

The next result shows how the use of the adaptive β_{k+1} given in (15) can ensure that (21) holds by taking w to be a sufficiently large positive number.

Theorem 4 *Assume $U + U^T$ is positive-definite and w is sufficiently large. In this case a sufficient condition for monotonic convergence is that*

$$w > \|e_0\|^2\Big(\frac{\sigma_{max}(U^TU)}{\sigma_{min}(U+U^T)} - 1\Big) \quad (24)$$

where $\sigma_{max}(U^TU)$ is largest eigenvalue of the matrix U and $\sigma_{min}(U + U^T)$ is the smallest eigenvalue of the matrix $U + U^T$.

Proof 3 *Substituting (15) into inequality (23) with a couple of algebraic manipulations gives a necessary and sufficient condition for monotonic convergence*

$$w > \frac{\|e_k\|^2 e_k^TU^TUe_k}{2e_k^TUe_k} - \|e_k\|^2 \quad (25)$$

Making the two estimates

$$\sigma_{max}(U^TU)\|e_k\|^2 \geq e_k^TU^TUe_k \quad (26)$$

and

$$\sigma_{min}(U + U^T)\|e_k\|^2 \leq 2e_k^TUe_k \quad (27)$$

a the sufficient condition for convergence becomes

$$w > \|e_k\|^2\Big(\frac{\sigma_{max}(U^TU)}{\sigma_{min}(U+U^T)} - 1\Big) \quad (28)$$

Since the initial guess u_0 results in a bounded tracking error e_0 then for any w such that inequality (28) holds, i.e. a w that ensures $\|e_0\| \geq \|e_1\|$, then inequality (28) will also hold for $e_k = e_1$. Inductively condition (24) holds for an arbitrary iteration k and therefore convergence is monotonic.

As noted in the previous section, for the nominal case $G_e = G_o$ the selection of $w = w_1\|e_k\|^2$ in fact gives geometric error convergence. The next proposition extends this to the case where G_e has positive multiplicative uncertainty.

Theorem 5 *If $U + U^T$ is a positive-definite matrix and $e_k \neq 0$ then there exists a w such that $\|e_{k+1}\| \leq \alpha\|e_k\|$ where $0 \leq \alpha < 1$.*

Proof 4 *The choice of* $w = w_1\|e_k\|^2$ *yields the following equation for* $\|e_{k+1}\|^2$

$$\|e_{k+1}\|^2 = \|e_k\|^2 - \gamma 2 e_k^T U e_k + \gamma^2 e_k^T U^T U e_k \quad (29)$$

where $\gamma = (1 + w_1)^{-1}$. *Note that since* $U + U^T$ *is positive-definite then the second and third right-hand terms in (29) are strictly negative and strictly positive respectively for an arbitrary* $e_k \neq 0$. *Using the estimates in (26) and (27) gives*

$$\|e_{k+1}\|^2 \leq \alpha^2 \|e_k\|^2 \quad (30)$$

where

$$\alpha^2 = 1 - \gamma\sigma_{min}(U + U^T) + \gamma^2\sigma_{max}(U^T U) \quad (31)$$

Since the negative term $-\gamma\sigma_{min}(U + U^T)$ *is of* $O(1+w_1)^{-1}$ *and the positive term* $\gamma^2\sigma_{max}(U^T U)$ *is of* $O(1+w_1)^{-2}$ *then by using a sufficiently large* $w_1 > 0$ *it is ensured that* $0 \leq \alpha < 1$, *giving geometric convergence.*

5. Experimental Results

A multi-axis test facility has been constructed so as to practically test ILC on a wide range of dynamic systems in an industrial-style application. Currently, the apparatus consists of a three-axis gantry robot supported above one end of a 6m long industrial plastic chain conveyor. A description of the test facility can be found in [6]. A 100Hz sample frequency was used to calculate the inverse models for each axis. The combined displacement reference trajectories for each axis (Figure 1) produce a 'pick and place' action, designed to collect a payload from a dispenser, synchronise position and velocity with the conveyor and place the payload on the conveyor. The reference trajectories define the iteration time period as 2 seconds. With a 100Hz sample frequency, this results in 200 sample instants per iteration.

The inverse algorithm has been implemented with a range of values for β, in order to experimentally verify the algorithms performance. Space limitations preclude a preclude a comprehensive discussion of the results but the effect of β on limiting performance in the presence of measurement noise is included as an illustration. This has been investigated by deliberately adding

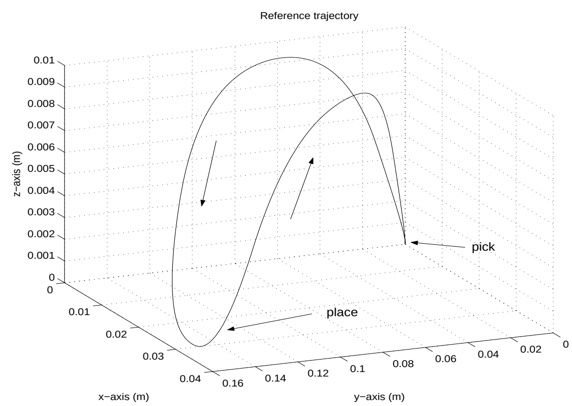

Figure 1. 3D reference trajectories

bounded, zero-mean, pseudo-random noise to the axis displacement signal recorded from the test facility by optical incremental encoders. The noise is pseudo-random, because it is generated by a seeded random number generator. In these experiments, the seed is the product of the sample number and the iteration number. Therefore for different iterations, the added noise appears to be random. However, for different tests, the same value of noise is added for corresponding samples during corresponding iterations.

Figure 2 displays the mean squared tracking error (mse) on a logarithmic scale (in mm^2) recorded for each iteration, with learning gain β equal to 0.1, 0.2 and 0.3. The pseudo-random noise has maximum bounds specified as ± 0.1mm. Similar results were obtained for the Y and Z axes and hence the plots have been omitted here. Previous experimental work has shown that the inverse algorithm is sensitive to the combination of measurement noise and high-frequency nonlinearities. The noise builds up in the iteration loop due to high-frequency nonlinearities, rapidly corrupting the plant input signal and causing much degraded performance.

The figures clearly demonstrate that convergence speed is proportional to β whereas minimum mse attained is inversely proportional to β, i.e. a trade-off is evident. This analysis is some-

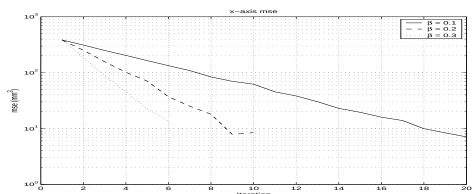

Figure 2. X-axis mse with $\beta = 0.1$, 0.2 and 0.3 ($\pm$0.1mm bounded noise)

what biased by how many iterations the algorithm can perform before the noise becomes sufficiently large to force a system shutdown. Therefore it is necessary to develop a test where the control system converges to minimum error and remains stable.

6. Conclusions

The use of inverse type ILC in the past has been unpopular owing to the belief that it lacks robustness. This paper however, shows that if an adaptive learning gain is added to the algorithm the system will geometrically converge if the plant multiplicative uncertainty satisfies a positivity condition. The adaptive learning gain is selected by the optimisation of an objective function that balances the reduction of the tracking error with the size of the learning gain. As a new theoretical result it has been shown that by decreasing the learning gain it is possible to achieve satisfactory tracking accuracy under the presence of measurement noise. This new result has been validated on an industrial-scale gantry robot system.

It is also possible to extend the approach here to the case when noise needs to be considered in design. Moreover, there are as yet unexplored, at the experimental level, relations between the approach here and that in, for example, [7]. This is the subject of ongoing work.

REFERENCES

1. S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operations of robots by learning", Journal of Robotic Systems, vol. 1, pp. 123-140,1984.
2. D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control (a learning-based method for high performance tracking control), IEEE Control Systems Magazine, 26(3), pp. 96–114, 2006.
3. M. Norrlof "An adaptive iterative learning control algorithm with experiments on an industrial robot", IEEE Transactions on Robotics and Automation, 18(2), pp. 245–251, 2002.
4. J. Hatonen, D.H. Owens, and K.L. Moore, "An Algebraic Approach to Iterative Learning Control", International Journal of Control, 7(1), pp. 45-54, 2004.
5. N. Amann, D.H. Owens, and E. Rogers, "Iterative learning control using optimal feedback and feedforward actions", International Journal of Control, 65(2), pp. 277-293, 1996.
6. J. D. Ratcliffe, P. L. Lewin, E. Rogers, J. J. Hatonen, and D. H. Owens,"Norm-optimal iterative learning control applied to gantry robots for automation applications, IEEE Trans Robotics, 22(6), pp. 1303-1307, 2006.
7. S. S. Saab, "A discrete-time stochastic learning control algorithm", IEEE Trans on Auto Control, 46(6) pp. 877-887, 2001.