Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004

Reducing residual vibrations through Iterative Learning Control, with application to a wafer stage

Casper L. van Oosten[1], Okko H. Bosgra[1,2] and Branko G. Dijkstra[3]

***Abstract* - This paper uses iterative learning control (ILC) to remove terminal residual vibrations. By carefully selecting the ILC observation and actuation windows, ILC brings the system to rest within a finite time interval using feedforward. Experiments conducted on an industrial high-precision set-up show that the vibrations can be removed well within half the period of the dominant vibration. Estimation of the vibration state from different experiments shows that ILC does indeed remove the residual vibrations. Analysis of the experimental results show that the feedforward profile obtained through ILC outperforms feedforward profiles obtained through simulation of LQ-state feedback and does not suffer from model uncertainty.**

INTRODUCTION

A high precision system that performs a motion from one position to another tends to exhibit vibrations caused by excitation of the system vibrational modes. These vibrations cause position errors after completion of the trajectory, effectively elongating the servo interval. A well known way to eliminate residual vibrations is by compensation of the dominant system modes, for instance with impulse input shaping [2][3][4][5][6][7]. In [1] it was shown that an iterative learning controller (ILC) is capable of removing these vibrations during the motion of the system by superimposing an additional feedforward signal on the original feedforward signal. This was however at the cost of increasing the maximum actuation force.

In this paper any modifications to the basic feedforward will be avoided by activating the ILC only on time intervals after the completion of the original motion. The feedforward will only be active on the interval t_0 to t_1, while the controller only observes the interval t_2 to t_3 (Figure 1). Different choices of the observation and actuation intervals are possible and intervals can overlap. In this paper, t_0 is defined as the end of the initial feed forward and t_3 the end of the servo interval.

1. Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands
2. Department of Mechanical Engineering, Eindhoven University of Technology, Den Dolech 2, 5800 ME Eindhoven, The Netherlands, E-mail: O.H.Bosgra@TUE.nl
3. Mapper Lithography BV, Lorentzweg 1, 2628 CJ Delft, The Netherlands

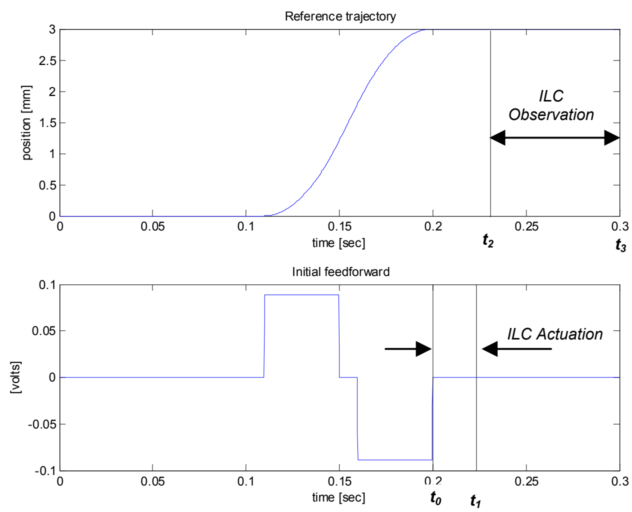

Figure 1: reference trajectory with actuation and observation intervals for ILC

It will be shown that ILC is capable of removing residual vibrations well within half the period of the dominant system vibration. Furthermore, it also effectively removes vibrations caused by unmodeled system dynamics.The technique will be illustrated with an application to an industrial high precision wafer-stage and the paper will analyze the mechanism behind the actual residual vibration.

ITERATIVE LEARNING CONTROLLER DESIGN

The iterative learning controller is designed in a similar way as in [1]. For a closed loop system shown in Figure 2, the mapping from a finite time interval input signal u to a

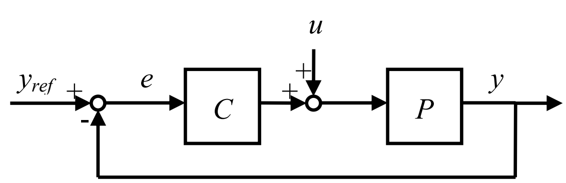

Figure 2: Closed loop servo control system

finite time interval output signal y is defined by the impulse response matrix H. Similarly, the mapping from reference y_{ref} to the tracking error e is denoted by the matrix S. Please refer to [8][9][10] for more details on this description.

0-7803-8335-4/04/$17.00 ©2004 AACC

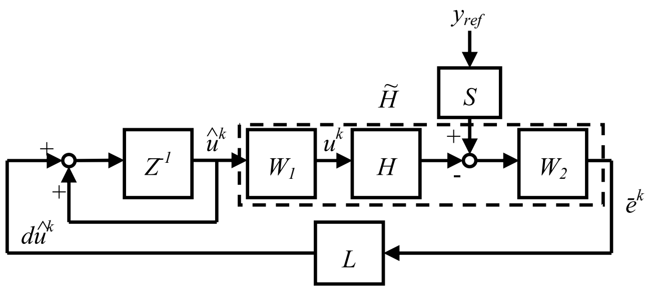

Figure 3: Iterative Learning Control scheme with weightings

In order to force the ILC to be active relative to the actuation and observation intervals as indicated in Figure 1, the ILC scheme from [1] is modified by adding weightings matrices W_1 and W_2, as shown in Figure 3. These weighting matrices have the form

$$W_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad W_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & I_{33} \end{bmatrix} \tag{1}$$

where the size of I_{22} and I_{33} determines the actuation and observation intervals respectively. These weighting matrices define a new process sensitivity matrix $\widetilde{H}$ given by

$$\begin{aligned} \widetilde{H} &= W_2 H W_1 \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & I_{33} \end{bmatrix} \begin{bmatrix} H_{11} & 0 & 0 \\ H_{21} & H_{22} & 0 \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & I_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I_{33} H_{32} I_{22} & 0 \end{bmatrix} \end{aligned} \tag{2}$$

The learning matrix L is the interconnection between the error and the update of the next trials' input. This feedback matrix is designed using an LQR design with weightings $Q=I$ and $R=\beta$. The solution is presented for example in [8] and is given by:

$$L = (\widetilde{H}^T \widetilde{H} + \beta I)^{-1} \widetilde{H}^T \tag{3}$$

The error signal $\bar{e}^k$ is denoted with an overbar because it only has non-zero elements for the time range t_2 to t_3. Similarly, the feedforward signal and its update, $\hat{u}^k$ and $d\hat{u}^k$ only have non-zero entries for the elements corresponding to the interval t_0 to t_1. Because of the nice structure of L (similar to the structure of $\widetilde{H}$) the computation of $d\hat{u}^k$ is numerically not very demanding. However, if computational load prohibits fast computation of $d\hat{u}^k$ a less demanding method is presented in [9] where $d\hat{u}^k$ is found by simulation of a Hamiltonian system.

EVALUATION OF THE FEEDFORWARD SIGNAL

In this section the goal is to evaluate the performance of the feedforward found with the iterative learning controller. As a first step, the initial states are estimated from experiments with different actuation and observation windows using an experimental frequency response fit state-space model of the system. If the same initial condition $x(t_0)$ is found for all experiments, this indicates that the motion removed by the feedforward is indeed the residual vibration caused by this initial condition.

To find the initial condition $x(t_0)$, assume the system to be linear time invariant. Furthermore, let an experiment with measured feedforward u and output y be given and let x_0 be the unknown initial condition at time t_0. Then the trajectory of the output y from t_0 to t_3 is the result of the initial state x_0 and input u. The trajectory caused by the output, denoted by y_0, is obtained by simulating the system using a state-space model of the system:

$$\begin{aligned} x(n+1) &= Ax(n) + Bu(n), \;\; x(0) = 0 \\ y_u(n) &= Cx(n) \end{aligned} \tag{4}$$

$$y_0(n) = y(n) - y_u(n) \tag{5}$$

The initial condition can be found by solving the optimization problem

$$\min_{x_0 \in \mathbb{R}^N} \left\| \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{N-1} \end{bmatrix} [x_0] - \begin{bmatrix} y_0(0) \\ y_0(1) \\ \vdots \\ y_0(N) \end{bmatrix} \right\|_2 \tag{6}$$

Second, a competitive feedforward is calculated. For a discrete time state-space system (4), a feedforward signal can be calculated using purely algebraic equations. The output of the system (4) is determined by the initial state x_0 and input ff. The problem solved by the point-to-point ILC can be written as follows

$$\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_i \\ \hline y_{i+1} \\ \vdots \\ y_N \end{bmatrix} = \left[\begin{array}{cccc|ccc} 0 & & & & & & \\ CB & 0 & & & & & \\ \vdots & \ddots & 0 & & & & \\ CA^{i-1}B & \cdots & CB & 0 & & & \\ \hline CA^{i}B & \cdots & \cdots & CB & 0 & & \\ \vdots & & & \vdots & \ddots & 0 & \\ CA^{N-1}B & \cdots & \cdots & CA^{N-i-1}B & \cdots & CB & 0 \end{array}\right] \begin{bmatrix} ff_0 \\ ff_1 \\ \vdots \\ ff_i \\ \hline 0 \\ \vdots \\ 0 \end{bmatrix} + \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{i} \\ \hline CA^{i+1} \\ \vdots \\ CA^{N} \end{bmatrix} x_0 \tag{7}$$

The values $y_1, \ldots, y_i$ are not relevant from a point-to-point control objective point of view, thus the system (7) can be reduced to

$$\begin{bmatrix} y_{i+1} \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} CA^iB & \cdots & CB \\ \vdots & \ddots & \vdots \\ CA^{N-1}B & \cdots & CA^{N-i-1}B \end{bmatrix} \begin{bmatrix} ff_0 \\ \vdots \\ ff_i \end{bmatrix} + \begin{bmatrix} C \\ \vdots \\ CA^i \end{bmatrix} x_0 \quad (8)$$

$$= \hat{H}\hat{ff} + \Theta x_0$$

where $\hat{H}$ is the left bottom part of the Toeplitz matrix in equation (7) and Θ the observability matrix. Let the system (4) have m states. Furthermore, assume the system (4) to be of minimal order and fully observable, i.e. $\hat{H}$ and Θ have rank m. Now the goal is to find a ff such that $y_j=y_{ref}$ for $j=i+1, \ldots, N$. Define

$$\hat{e} := \begin{bmatrix} e_{i+1} \\ \vdots \\ e_N \end{bmatrix} = \begin{bmatrix} y_{ref} \\ \vdots \\ y_{ref} \end{bmatrix} - \begin{bmatrix} y_{i+1} \\ \vdots \\ y_N \end{bmatrix} \quad (9)$$

In practical applications there is a bound on the input. The balance between input and output will here be realized using a quadratic cost function, weighing the output error against the input effort. This is the finite time LQ problem. With weight $\hat{Q}$ on the error energy and weight $\hat{R}$ on the input effort, the optimal input ff is given by solving

$$\min_{ff \in \mathbb{R}^N} \left(\hat{e}^T \hat{Q} \hat{e} + ff^T \hat{R} ff \right) \quad (10)$$

Similar to the LQ-ILC weightings, the choice of $\hat{Q}=I$ and $\hat{R}=\beta I$ will enable to find a trade-off between input effort and output energy. In effect, the LQ-ILC solution is the solution to the problem (10). The LQ learning controller L minimizes the objective:

$$\min \sum_{k=1}^{\infty} y_k^T Q y_k + \Delta ff_k^T R \Delta ff_k \quad (11)$$

For the case that $y_{ref}=0$, the ILC solution is exactly equivalent to the solution of (10): For $k\rightarrow\infty$, the controller that minimizes the objective (11) yields the ILC solution that minimizes (10). The solution to equation (10) can be found using linear matrix equations. The learning controller thus solves a set of linear matrix equations in an iterative way. Under the assumptions for the system (4), the solution to (10) is given by

$$\left(\hat{H}^T \hat{Q} \hat{H} + \hat{R} \right) ff = \hat{H}^T \hat{Q} \Theta x_0 \quad (12)$$

The deadbeat solution is given for the case that $\beta=0$, i.e. $\hat{R}=0$. Let the system (4) have m states, rank $\hat{H}^T \hat{Q} \Theta = m$ and rank $\left(\hat{H}^T \hat{Q} \hat{H} + \hat{R} \right) = m$. The solution ff is thus exact when the length of the input ff equals the number of states m.

Finally, the ILC performance can be evaluated by applying these results to the system and comparing them to the results obtained with ILC.

EXPERIMENTAL RESULTS

The method described above has been used to eliminate oscillations in one direction of a high precision XY

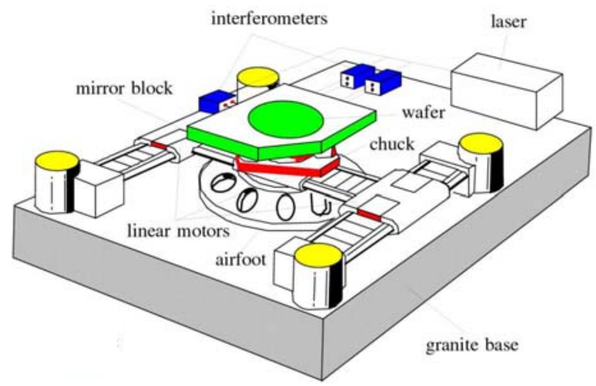

Figure 4: Wafer stage of a wafer stepper
Picture appearing courtesy of D. de Roover

positioning table of a wafer stepper, a wafer stage (Figure 4). For a wafer stage, the cycle time is defined as the time to move from one position to another, plus the time it takes to settle within the desired boundary of ±100*nm*. The reference trajectory and initial feedforward used in the experiments are shown in Figure 1, where t_0 is 0.2 seconds and t_3 is 0.3 seconds. The designed step time for this motion is 0.0904 seconds. Using only the basic feedforward trajectory, it takes an additional 0.02 seconds for the system to settle within the accuracy bounds, yielding a cycle time of 0.1104 seconds (Figure 5).

If ILC is applied to the whole interval t_0 to t_3, the system

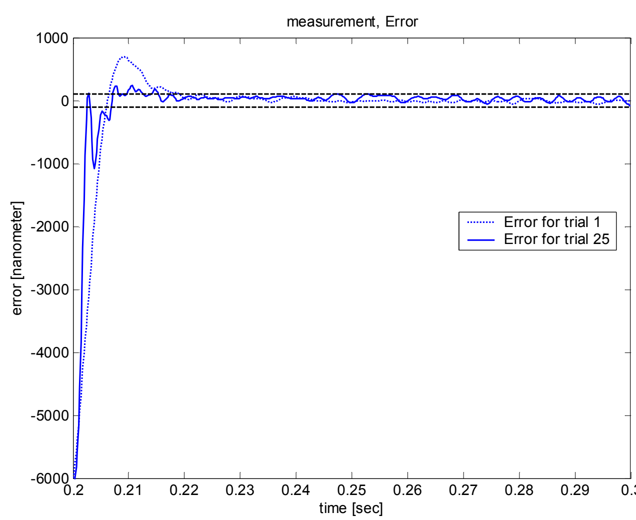

Figure 5: vibration error, trial 1 and trial 25, overlapping observation and actuation windows, [0.2 : 0.3]

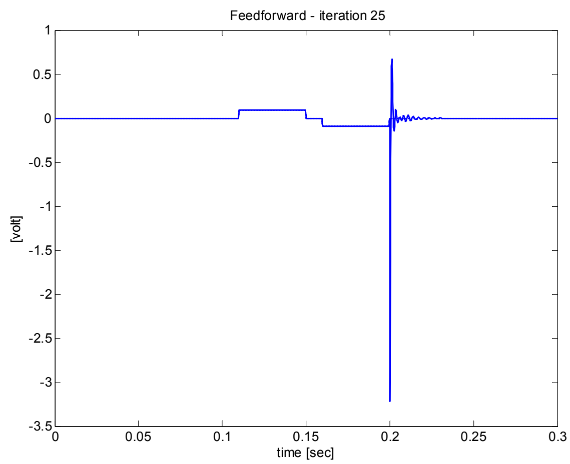

Figure 6: feedforward, trial 25 overlapping observation and actuation windows, [0.2 : 0.3]

settles between the accuracy bounds slightly faster in 0.0155 seconds (Figure 5). This is at the costs of an unacceptably large feed forward signal (Figure 6), caused by the ILC efforts to reduce the error at the beginning of the interval.

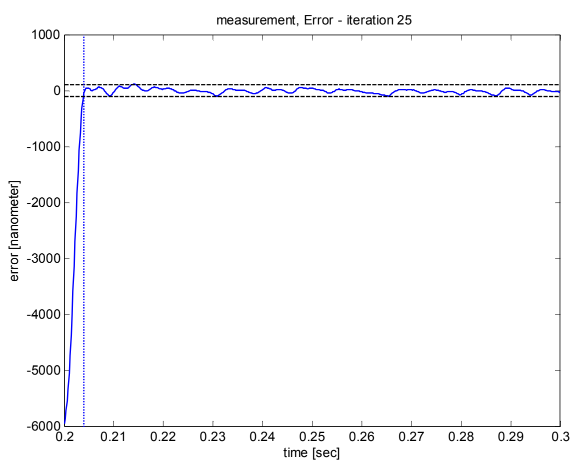

Figure 7: vibration error, trial 25, observation interval [0.204 : 0.300]

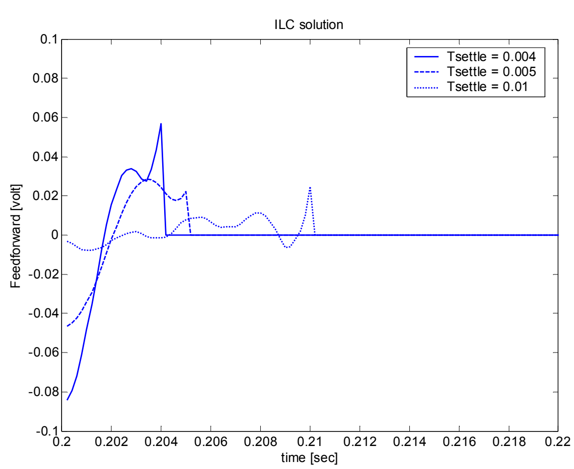

Figure 8: zoom in on feedforward, trial 25, for three different actuation intervals

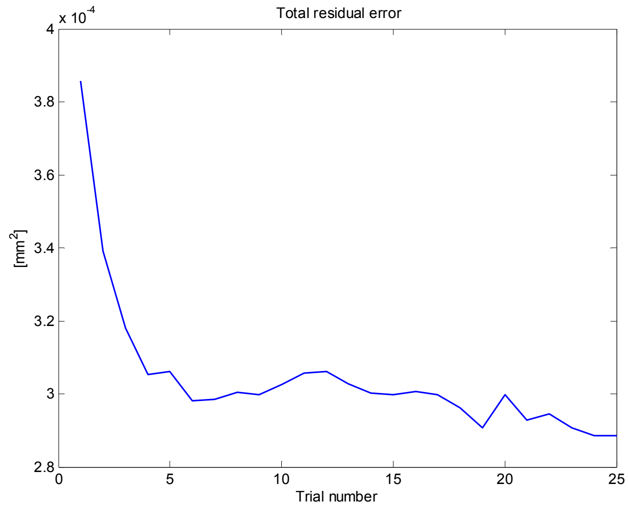

Figure 9. Sum of squared residual error, trial 1 – 25, actuation window [0.200 : 0.204]

To avoid this large feedforward signal, both the observation and actuation intervals have been adjusted such that they do not overlap, with the end of the actuation interval marking the beginning of the observation interval (t_1=t_2). After all, the system is required to be at rest during the observation interval, so no correcting feedforward should be necessary. Several experiments were performed, where the actuation interval was shortened consecutively. The shortest settling time was obtained by setting t_1=t_2=0.204 seconds, forcing the vibration error to be within bounds within that same interval (Figure 7). The feedforward for several settling times is shown in Figure 8. The shortest feedforward removes 80% of the settling time and reduces the cycle time by 14%! The feedforward profile found this way is still below the maximum value of the basis feedforward. Although the ILC needs 25 trials to find a feedforward that does this, the largest part of the residual error is already removed within the first 5 trials (Figure 9). This figure shows the sum of the squared error. At trial 25, the largest part of this sum is caused by the error of the first 20 samples.

EVALUATION OF ILC EXPERIMENT RESULTS

The results will be evaluated by comparing ILC results with the results from a finite time LQ problem. Using the experiments with t_1=t_2=0.204, t_1=t_2=0.210, t_1=t_2=0.220 and no actuation, the initial condition $x(t_0)$ was estimated for each signal pair using a 32[nd] order model of the system, based on experimental system identification. Figure 10 shows that the initial states can be estimated consistently from the four experiments. For all experiments the estimation is based on the first 20 samples, starting at t_0. The residuals from the optimization in equation (6) are plotted in figure 10 and show that the estimation error is very small compared to the output trajectories as in Figures 5, 6 and 8. The motion removed by iterative learning control is indeed the residual vibration quantified by the initial condition $x(t_0)$.

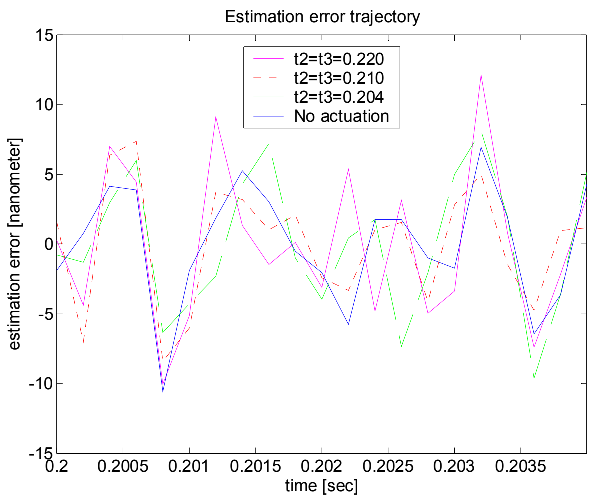

Figure 10: residuals of estimation of initial condition from different experiments.

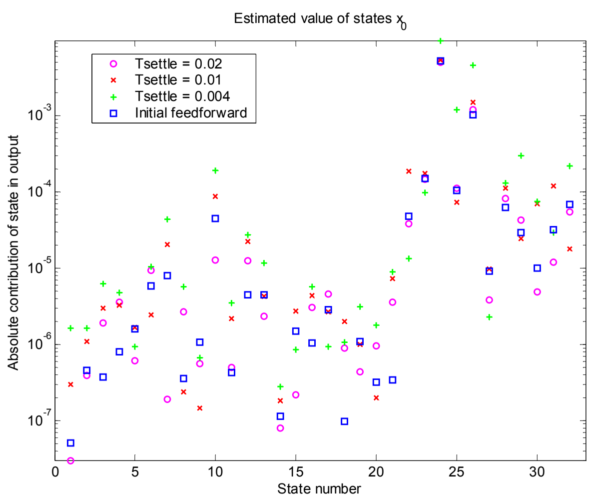

Figure 11: estimated value of reduced model states in modal domain at t_0, scaled by output matrix *C*.

Using the state-space model, more information is available about the behavior of the system: the dominant mode in the residual vibration is the mode that belongs to states 23 and 24 in Figure 10. This mode has undamped period T_n = 0.0142 seconds and damping coefficient ζ = 0.585. The settling time of 0.004 obtained by ILC is thus well below half the period of this mode.

With the use of the model and the initial condition, a feedforward profile is obtained by solving a finite time LQ problem. The initial condition estimated from the experiment with no actuation was chosen as initial condition for the simulation. To find a comparable feedforward, the settle time was fixed at the same intervals as the experiments: 4, 5 and 10 milliseconds. Then, the input weight β was adjusted such that the desired feedforward is obtained. The result is shown in Figure 12. The strong resemblance with the ILC result (Figure 9) confirms that the ILC solution and the finite time LQ problem are equivalent.

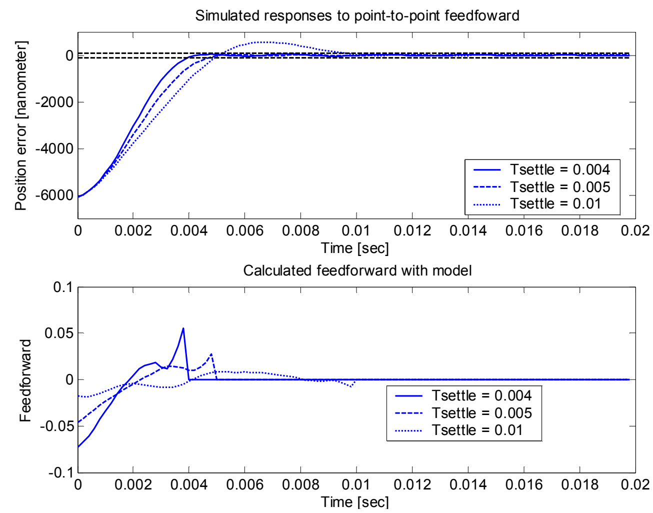

Figure 12. Simulated responses to finite time LQ-feedforward, actuation interval 4, 5 and 10 milliseconds

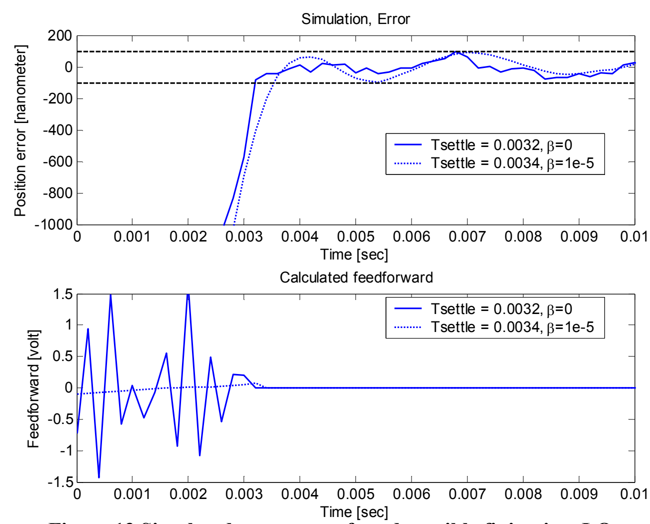

Figure 13 Simulated response to fasted possible finite time LQ-feedforward, with and without input weight

To check whether the settling time can be reduced any further, the deadbeat solution (β=0) will be used as a starting point. Because the order of the system model is 32, the deadbeat time is 32 samples or 6.4 milliseconds. The deadbeat feedforward however exceeds the maximum allowed feedforward of 0.1 volt and is therefore not useful in practice.

As a next step, the accuracy bounds on the settling and the maximum bounds on the feedforward are maxim, yielding a settling time of 3.2 milliseconds for an unconstrained feedforward and 3.4 milliseconds for a constrained feedforward (Figure 13). This solution is slightly faster then the ILC solution.

When this feedforward is applied to the system, the simulation results are not obtained (Figure 14). One apparent source for the deviation is the deviation in the

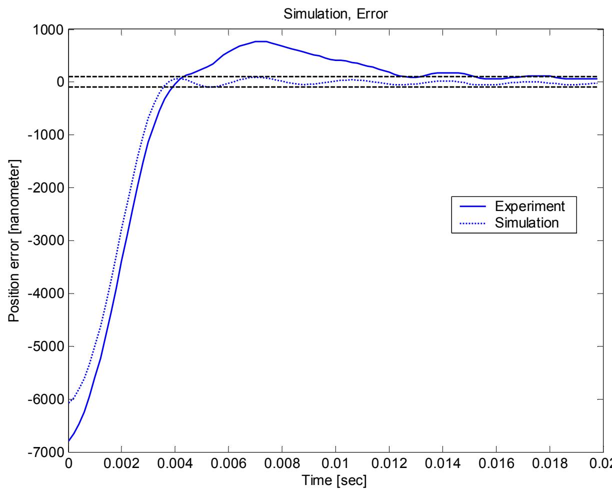

Figure 14: feedforward and output from LQ-state feedback compared to experimental ILC results

initial condition x_0. The experiments from which the initial condition was estimated were conducted on a different day. Variations in room conditions prevented the exact replication of the initial condition. The calculated feedforward is thus not very robust against varying dynamics. ILC offers a way to find a new feedforward within a few experiments and this feedforward can directly be applied in operations.

With a settle time of 5 milliseconds, the experimentally obtained ILC solution is near the simulated minimum settle time of 3.4 milliseconds. Iterative learning control thus offers a competitive technique for solving a point-to-point solution. The experiment-based nature and straightforward application of ILC allows a fast and reliable update of a feedforward signal. In this way, it can deal with the varying system dynamics that the waferstepper suffers from.

CONCLUSIONS

This paper shows that iterative learning control has the potential to reduce the cycle time caused by residual vibrations in a point-to-point motion to almost one-third the period of the dominant vibration, while leaving the original feedforward profile intact. Time weightings on the input and output of the controller force the ILC to a feedforward profile that brings the system to rest within the given finite time interval. The obtained feedforward profile has limited amplitude and removes any overshoot in the output. The initial condition at the beginning of the time interval can be estimated consistently using a model of the system. This shows that iterative learning control indeed removes the residual vibrations caused by this initial condition. The ILC result is very close to the minimum possible settle time. Iterative learning control is a fast and reliable way to obtain a high level of active suppression of residual vibration.

REFERENCES

[1] Branko G. Dijkstra and Okko H. Bosgra. Exploiting Iterative Learning Control for Input Shaping, with application to a wafer stage. *American Control Conference,* June 2003.

[2] N.C. Singer and W.P. Seering. Preshaping command inputs to reduce systems vibration. *Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control*, 112:76-82, 1990.

[3] P.H. Meckl and W.P. Seering. Experimental evaluation of shaped inputs to reduce vibration for a cartesian robot. *Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control*, 112:159-165, June 1990.

[4] W.E. Singhose and N.C. Singer. Effects of input shaping on two-dimensional trajectory following. *IEEE Transactions on Robotics and Automation*, 12(6):881-887, 1996.

[5] Roberto Kinceler and Peter H. Meckl. Corrective input shaping for a flexible-joint manipulator. *American Control Conference*, pages 1335-1339, June 1997.

[6] Edwin Hou Lucy Y. Pao, Timothy N. Chang. Input shaper designs for minimizing the expected level of residual vibration in flexible structures. *Proceedings of the American Control Conference*, pages 3542-3546, 1997.

[7] Marc Bodson. An adaptive algorithm for the tuning of two input shaping methods. *Automatica*, 34(6):771-776, 1998.

[8] Branko G. Dijkstra and Okko H. Bosgra. Extrapolation of optimal lifted system ILC solution, with application to a waferstage. *American Control Conference*, May 2002.

[9] Branko G. Dijkstra and Okko H. Bosgra. Noise suppression in buffer-state iterative learning control, applied to a high precision wafer stage. *Conference on Control Automation*, September 2002.

[10] Branko G. Dijkstra and Okko H. Bosgra. Convergence design considerations of low order q-ILC for closed loop systems, implemented on a high precision wafer stage. *Conference on Decision and Control*, December 2002.

[11] Casper van Oosten. *Robustness for Iterative Learning Control, with application to a wafer stage*. MSc. Thesis, Delft University of Technology, December 2003.

[12] Rob Tousain and Eduard Van der Meché. Design strategies for iterative learning control based on optimal control. *Conference on Decision and Control*, 40:4463-4468, December 2001.

[13] Won Cheol Kim, In Sik Chin, Kwang Soon Lee, and Jinhoon Choi. Analysis and reduced-order design of quadratic criterion-based iterative learning control using singular value decomposition. *Computers and Chemical Engineering*, 24:1815-1819, 2000.