Standard and lifted approaches of iterative and learning control applied on a motion system

STANDARD AND LIFTED APPROACHES OF ITERATIVE LEARNING CONTROL APPLIED ON A MOTION SYSTEM

Iuliana Rotariu [*,1] **Branko Dijkstra** [**] **Maarten Steinbuch** [***]

[*] The University of Sydney
Australian Centre for Field Robotics
The Rose Street Building J04, NSW 2006 Australia
[**] Delft University of Technology, Mechanical Engineering, System and Control Group, Mekelweg 2, 2628 CD Delft, The Netherlands
[***] Eindhoven University of Technology, Mechanical Engineering, Control Systems Technology, P.O. Box 513, WH 0.141, 5600 MB, Eindhoven, The Netherlands

Abstract: Iterative Learning Control (ILC) is a technique for improving the performance of systems or processes that operate repetitively over a fixed time interval. The basic idea of ILC is that it exploits every possibility to incorporate past repetitive control information, such as tracking errors and control input signals into the construction of the present control action. Past control information is stored and then used in the control action in order to ensure that the system meets the control specifications such as convergence. The goal of the research presented in this paper is to liken two different ILC techniques applied to the wafer stage of a wafer scanner motion system. Namely, we consider briefly the concepts of *standard* and *lifted* ILC and we evaluate the ILC performance in terms of tracking errors.

Keywords: Control, Learning, Motion system

1. INTRODUCTION

Iterative Learning Control (ILC) is a technique for improving the performance of systems or processes that operate repetitively over a fixed time interval, see for instance (Bien 1998). The basic idea of ILC is that it exploits every possibility to incorporate past repetitive control information, such as tracking errors and control input signals into the construction of the present control action. Past control information is stored and then used in the control action in order to ensure that the system meets the control specifications such as convergence.
ILC might be seen as a technique to generate a feedforward signal effective for providing good tracking control.

The application we consider is a wafer scanner motion system. Wafer scanners are opto-mechanical machines for producing Integrated Circuits (ICs) on a silicon wafer using a photolithographic process. One of the main components of a wafer scanner is the six degrees of freedom (DOF's) wafer stage (Wal 2002). This is an electromechanical servo system that positions the wafer (200-300 mm diameter) with respect to the imaging optics. The wafer stage largely determines the throughput (80-100 wafers/h, 80-200 ICs/wafer) and the accuracy of the products is subject to severe performance requirements. Normal scan speeds and accelerations are 0.5 m/s and 10 m/s^2, respectively. The positioning accuracy is in terms of nanometers and microradians. Such high accuracy is needed for the fine patterns to be produced (typical dimension: 180-250 nm). Moreover, various layers (typically: 20) of different patterns have to be aligned very accurately with respect to each other.

[1] Research performed while working for Philips Center for Industrial Technology (CFT), Eindhoven, The Netherlands

To achieve the demanding specifications, a sound mechatronic design is very important. The design of mechanics, electronics, software, and control must go hand-in-hand. The performance may be limited by sensor noise, actuator dynamics and constraints, sample rate limitations, control structure restrictions, or hard nonlinearities like Coulomb friction and backlash. Nowadays, wafer stages move almost frictionless due to air bearings. Modern theory on systems and control offers a large number of (different) techniques for designing a high-performance motion control system, like $\mathcal{H}_\infty$ feedback control, Iterative Learning Control, and many others. The goal of the research presented in this paper is to compare two different ILC techniques applied to the y direction of a wafer stage.

This paper is organized as follows: in Section 2 we describe in short standard ILC, we explain the limitations a fixed robustness filter in the standard design. We present the experimental results in terms of tracking errors while scanning on a wafer. In Section 3 we consider briefly the concept of lifted ILC and its performance in terms of tracking errors. Section 4 concludes the most important issues.

2. STANDARD ILC

In this section we will introduce the basic rules and control design of standard ILC. Figure 1 shows the *standard* ILC loop. We restrict the study to the case where the plant is a causal, LTI dynamical system G. C is a feedback controller which insures the stability of the closed loop system.
We suppose that the desired response r is defined on the interval (t_0, t_f), where $t_f \leq \infty$ and the initial conditions are the same at the beginning of each trial.

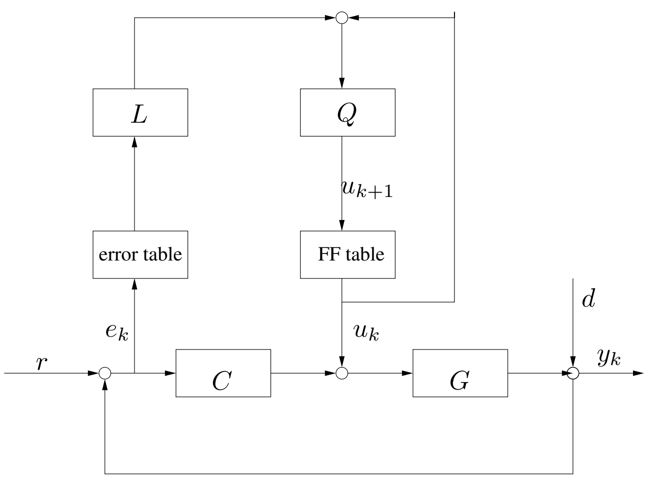

Fig. 1. Closed-loop LC configuration.

The goal of ILC design is to produce the signal u^* such that $r = Gu^*$. We seek a sequence of inputs u_k with the property that $\lim_{k\to\infty} u_k = u^*$, where the index k is the iteration trial. A prototype update law that implements iterative learning control by updating the past iteration input u_k on the basis of the past error is

$$u_{k+1} = u_k + Le_k, \quad (1)$$

where L is the learning filter to be designed. Taking into account that

$$e_k = Sr - P_s u_k, \quad (2)$$

where S and P_s are operators corresponding to the sensitivity ($S = \frac{1}{1+GC}$) respectively the process sensitivity ($P_s = \frac{G}{1+GC}$) function, the recursion for the input is given by

$$u_{k+1} = (I - LP_s)u_k + LSr. \quad (3)$$

Using the Fixed point theorem (Moore 1993), the sequence of inputs $\{u_k\}_k$ converges to a fixed point u^* if

$$\|I - LP_s\| < 1, \quad (4)$$

where $\|\cdot\|$ is any norm over a Banach space of operators. For example, we may consider $\mathcal{H}_\infty$ control theory for solving problems involving the $\mathcal{H}_\infty$ norm (Moore 1993). For LTI systems, the $\mathcal{H}_\infty$ convergence condition is equivalent (see (Moore 1993)) with the following frequency domain description:

$$|I - L(s)P_s(s)| < 1, \text{ for any s} \in j\mathbf{R}. \quad (5)$$

From (5) it follows easily that the optimal L filter is given by $L(s) = (P_s(s))^{-1}$. In this paper we show how to implement learning control for systems which are not necessarily invertible over the hole frequency band. The robustness filter Q will take care that the convergence criterion (5) is satisfied for all frequencies. We shall consider as a robustness filter Q a low-pass filter whom cut-off frequency is determined such that

$$|Q(s)(I - L(s)P_s(s))| < 1 \text{ for any } s \in j\mathbf{R} \quad (6)$$

holds and $Q(s) \approx 1$ when (5) already holds. The closed-loop learning control configuration is depicted in the Figure 1.

In order to design a learning controller for our experiments, the following steps will be followed:

- Plant modeling based on measured Frequency Response Function (FRF). For the y direction a model is determined using a polynomial fit routine.
- Design a feedback controller; the feedback compensators should be known and fixed over a sequence of ILC experiments.
- Design learning filter L and the robustness filter Q such that the convergence criterion (6) is valid.

As we have previously explained, the learning filter L has to approximate the inverse of the modeled

$P_s(s)$ function as good as it is possible. For a proper minimum phase modeled process sensitivity function, one can compute and implement its inverse without any problems. For non-minimum phase plants, a stable approximation of the real inverse is used. In this paper, in order to obtain a stable inverse of the process sensitivity function, the ZPETC algorithm is used (Tomizuka 1987).

For the wafer stage system as depicted in the Figure 2, we observe a good inverse of the process sensitivity function (Figure 3) up to the Nyquist frequency of 4000 [Hz].

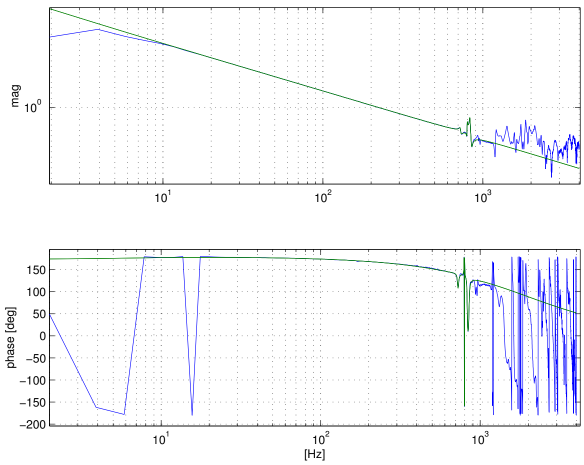

Fig. 2. The wafer stage test rig: The measured $\text{FRF}_{y \rightarrow y}(blue)$ and its approximation (green).

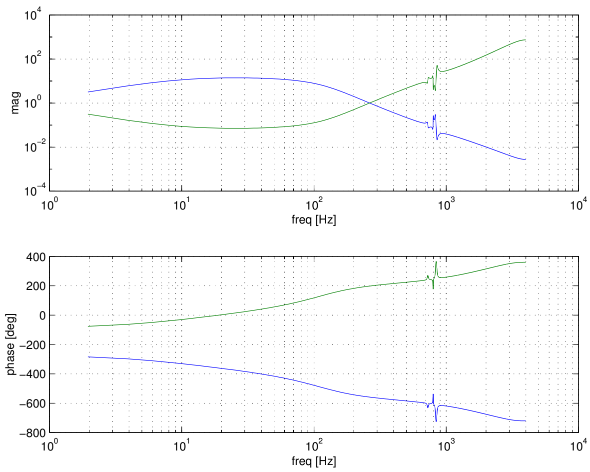

Fig. 3. The wafer stage test rig: P_s for the modeled plant and the 100 [Hz] bandwidth FB controller (blue) y. The inverse of the P_s function (green)

In the case of learning control implemented for y DOF of the short stroke of the motion system, we see that the inverse is reasonable compared to the modeled process sensitivity function up to a high frequency.

Since the learning filter is not able to ensure convergence, a robustness filter Q is necessary. In this paper we use as robustness filter a low pass Butterworth filter. The cut-off frequency depends on the learning controller : a good stable approximation of the inverse of the process sensitivity function up to a high frequency will allow a high cut-off frequency of the Q filter. Then the learning compensator will *learn* up to this high cut-off frequency.

The ILC is applied for a representative scan of velocity 0.5 [m/s], acceleration 5 [m/s^2], jerk 1000 [m/s^3] over 0.1 [m]. We first design a robust feedback controller (good stability margins) with respect to the variation of y dynamics along the scanning trajectory. The bandwidth of the feedback controller (0 dB crossing of the open loop) is about 100 [Hz]. We first design a high cut-off frequency of the Q-filter, namely at 1000 [Hz]. Notice that this frequency is ten times higher than the feedback bandwidth. We observe (see Figure 4) that the errors converge to zero around the position $(0, 0)$ (middle of time slot) but does not converge to zero at $(x, y) = (0, -25)$ (beginning of setpoint).

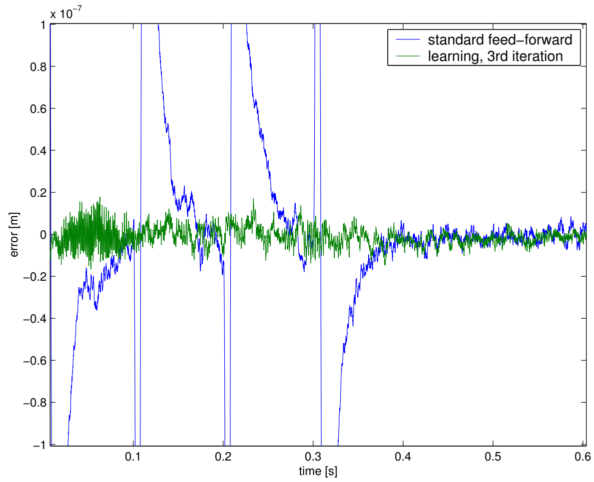

Fig. 4. Servo errors in the y direction cut-off frequency of the Q-filter=1000 [Hz]

We consider the convergence criterion plotted for measured y FRF at different points on the wafer stage $((0, 0), (0, -50), (0, -25), (0, 25), (0, 50))$ and for the modeled FRF at $(0, 0)$. Using the same ILC for all these positions for the y DOF, we see that convergence criterion is not satisfied (Figure 5).

Then we adapt adequately the cut-off frequency of the Q filter (Figure 6). Implementing ILC with robust stabilizing feedback controllers and position dependent convergent cut-off frequencies of the Q-filters, we obtain good tracking errors (Figure 7) for the y DOF.

Even during the acceleration phase we see very small errors.

3. LIFTED ITERATIVE LEARNING CONTROL

In this section we briefly present the lifted system representation and lifted ILC design setting (Dijkstra, June 2002). The system dynamics is considered as

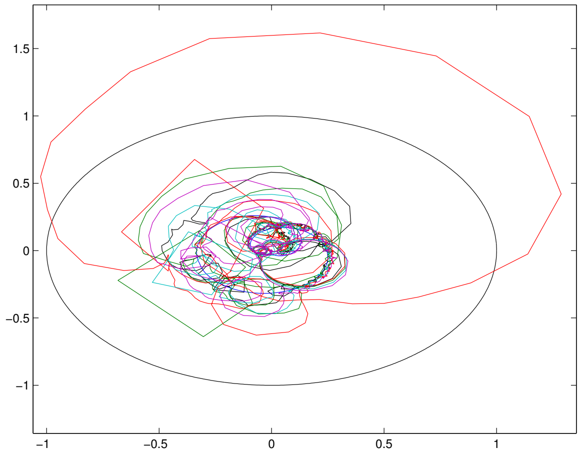

Fig. 5. Convergence criterion, y direction cut-off frequency of the Q-filter=1000 [Hz]

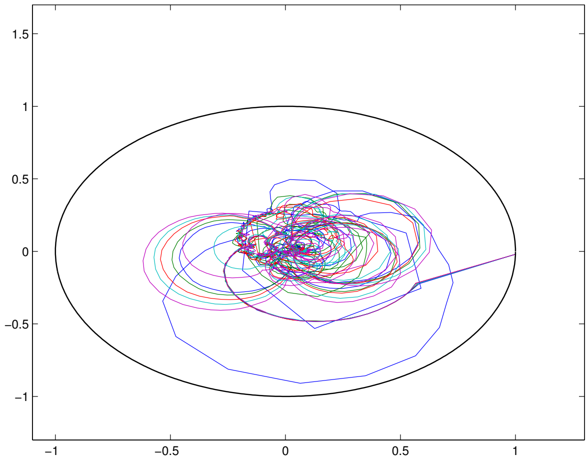

Fig. 6. Convergence criterion, y direction cut-off frequency of the Q-filter=750 [Hz]

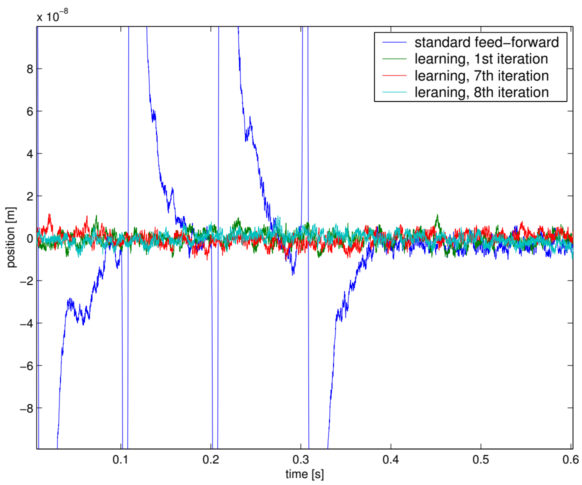

Fig. 7. Servo errors in the y direction cut-off frequency of the Q-filter=750 [Hz]

a static map, which describes the system behavior along a finite time interval. Lifted ILC will be used to obtain a feed-forward signal u that results in minimal tracking errors e with respect to the desired trajectory r. A general ILC-setup design for a single trial for lifted ILC (Dijkstra, September 2002) is shown in figure 8. The trajectory r to be followed is constant from trial to trial. The ILC controller L is the feedback interconnection between the error signal e_k and the input of the system x_k. All signals in Figure 8 are vectors of length N describing a finite discrete time signal. The blocks are system descriptions defined over a finite interval of length N.

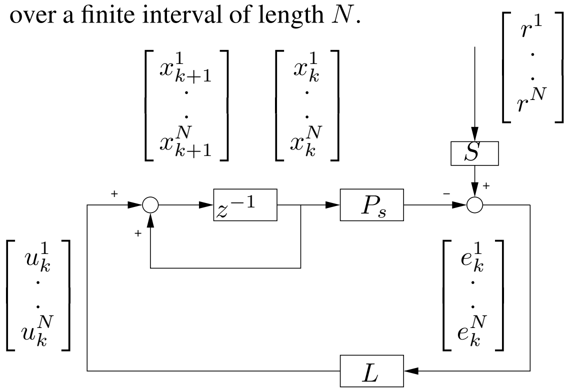

Fig. 8. Lifted ILC setting

In Figure 8, P_s is a matrix describing the finite time input-output map of the plant under consideration (the wafer stage stabilized by a feedback controller), $P_s = \frac{G}{1+GC}$.
For a causal LTI system, such a matrix will be a lower triangular Toeplitz matrix containing the Markov parameters of the LTI system (Dijkstra, June 2002). For a LTI system, the matrix P_s will be the input response of the process sensitivity function of the system. The principle of the ILC is that the measurement of the tracking error from the current trial e_k (k denotes the trial index of ILC) is used to connect the error to the next trial e_{k+1}. In Figure 8 this means that the error vector e_k (based on the input x_k and the reference r) is used to obtain the input for the next trial, namely the new learned feed-forward signal x_{k+1}. The block z^{-1} denotes one trial delay for the entire input vector x_k.

The goal of the ILC is to obtain a good error tracking e_k or, in other words, to eliminate the errors introduced by the reference signal r. By the internal model principle it follows that in order to eliminate the disturbances introduced by the reference signal r, the feedback controller of the plant P_s has to contain an integrator, shown with the feedback loop around delay z^{-1} which introduces one trial delay. L is the feedback gain matrix to stabilize the system. The controller L that stabilizes the system is a convergent ILC controller.

Therefore, the lifted ILC configuration translates the open loop design procedure of a learned feed-forward signal design into a feed-back control design problem. The filter L becomes a feedback controller which stabilizes the system given by the process sensitivity P_s function. The ideal ILC is, as in the case of standard ILC, the inverse of the P_s function.

Any feedback design method can be used for the design of L. Here, a LQR (linear-quadratic regulator)

design is considered with a linear quadratic objective to balance the outputs ($y_k = r - e_k$) and the inputs u_k. Such a balance gives exactly a trade-off between non systematic noise amplification and ILC convergence (Dijkstra, June 2002). The LQR solution is given as a closed-loop feedback control.
Let us consider the linear quadratic performance index

$$J = \sum_{k=1}^{M} y_k^T R_1 y_k + u_k^T R_2 u_k = \\ = \sum_{k=1}^{M} x_k^T (P_s)^T R_1 (P_s) x_k + u_k^T R_2 u_k. \quad (7)$$

Choose the control that minimizes the quadratic performance index (7). The control weighting R_2 and the state weighting R_1 are symmetric matrices chosen by the designer depending on control objectives. Weight matrix R_1 is assumed positive semi-definite ($R_1 \geq 0$) and R_2 is positive definite ($R_2 > 0$). Thus, J is always bounded below by zero and a sensible minimization problem results. Since the squares of the inputs respectively outputs (therefore implicitly the squares of the states of the system) occurs, the performance index (7) is a form of generalized energy and minimizing it will keep the states and controls small.

Important remarks for the next steps (we refer here to implementation algorithm) are:

- The system tends to be very large when long trajectories are considered.
- The matrix P_s may be a (nearly)singular matrix (when the underlying plant contains delays or non-minimum phase zeros). In this case the system is not fully observable, which is a problem for an optimal control solution. As solution for this problem, the Lifted ILC with SVD deviation and Lifted ILC with adjustment for observable part of P_s is considered (Dijkstra, June 2002), (Dijkstra, September 2002).

We consider now the solution of the optimal control problem (7) with $R_1 = I$ and $R_2 = \beta I$. An approximation for the solution of the corresponding Ricatti equation is given by

$$S = {P_s}^T P_s + \beta I, \quad (8)$$

with the feedback interconnection

$$LP_s = S^{-1} {P_s}^T P_s \quad L = S^{-1} {P_s}^T. \quad (9)$$

Remark. Substitution of this approximate solution in the Ricatti equation shows that this is a solution of an optimal control problem with a slightly different R_1, namely $R_1 = \left({P_s}^T P_s + \beta^2 I({P_s}^T P_s + 2\beta I)^{-1}\right)$. The fact that it is a solution of an optimal control problem means that it yields a stable solution and thus a convergent lifted ILC.

When using the lifted ILC solution (9) on a singular system matrix P_s, the singular values of P_s will be limited by β. Essentially, $P_s^T P_s$ may not be invertible but $P_s^T P_s + \beta I$ is invertible for any $\beta > 1$. This enables the method to be used even when the lifted ILC system is not fully observable (although it does result in some unused integrators in the ILC feedback).

The lifted ILC design has been shown to be able to handle both convergence and noise amplification in practical setting (Dijkstra, September 2002). The parameter β might be seen as a tuning parameter in order to control the balance between the tracking error performance and noise influence on the input signal. Intuitively, from formula (9) it follows that, for very small β, the ILC design will be more like a true inverse of the system (like standard ILC), and will thus not handle system noise very well.
Therefore, for small chosen β we expect a good and fast convergence of the learned errors but not a good noise performance (not better than in the case of standard ILC). For a bigger β, a nice balance between tracking error performance and noise performance can be achieved, but the ILC converges slower.

For $\beta = 0.01$ the learned errors do not converge. Error analysis reveals that 840 [Hz] and 1800 [Hz] frequencies destabilize the system.

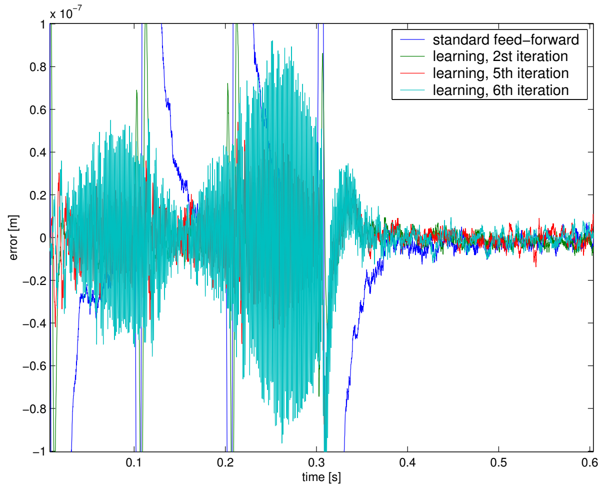

Fig. 9. Servo errors in the y direction, $\beta = 0.01$

The low-frequent noise/non-repetitive effects present in the learned errors (up to 250 [Hz]) is amplified. The servo error corresponding to the iteration trial $k = 6$ has everywhere a bigger amplitude than the servo errors at the previous iterations, see Figure 3.

For $\beta = 1$ a good tracking errors performance is achieved (Figure 10): the deterministic effects present in the tracking errors are learned and the low-frequent noise (up to 250 [Hz]) is not amplified.

For $\beta = 5$ still good tracking error performance is obtained (Figure 11). The convergence of the tracking errors is slower and low frequent noise and non-repetitive effects present in the learned errors are amplified.

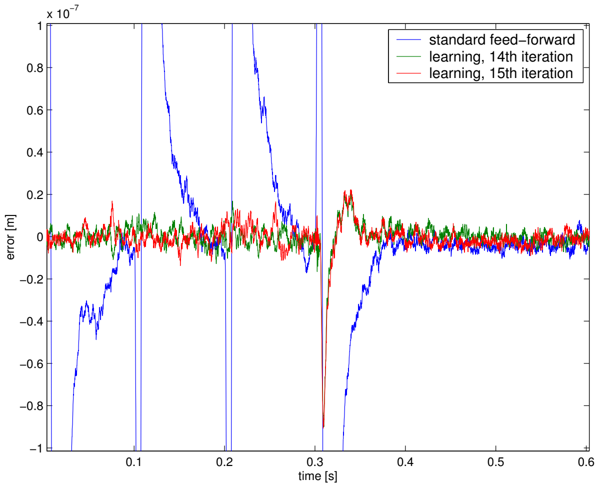

Fig. 10. Servo errors in the y direction, $\beta = 1$

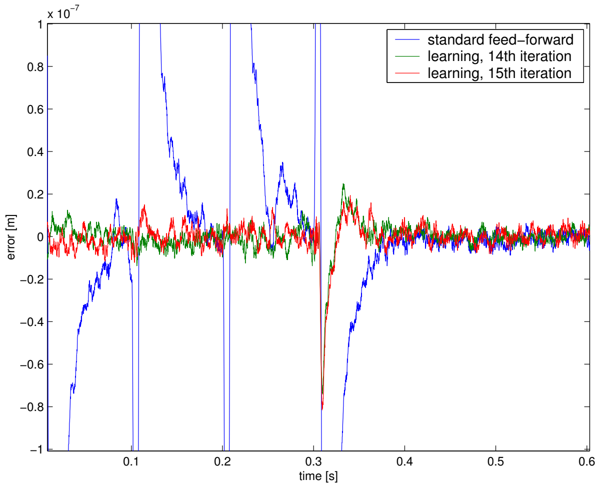

Fig. 11. Servo errors in the y direction, $\beta = 5$

4. CONCLUSIONS

- Applying standard ILC good tracking error along the entire scanning pattern is achieved. We have used a Q-filter in order to increase the robustness of the ILC against position-dependent dynamics, high-frequent noise and plant/model mismatch.
- For both ILC methods, higher order models would better identify relevant modes of the plant. This does not lead necessarily to the best ILC performance (optimal system identification from ILC viewpoint).
- For both ILC techniques, the learned feed-forward signal depends on the reference signal and it is sensitive to the setpoint trajectory changes.
- The convergence behaviour of the lifted ILC is determined by the choice of the weightings (the parameter β), which results in a trade-off between the input effort /noise amplification, and the tracking errors.
- For the standard ILC, the performance may be improved considering an adaptive filter in place of a fixed Q-filter: during the design of the Q-filter take into account the low/high frequency dynamics and noise present in the system. in the design of the Q-filter take into account if there exists high/low frequency dynamics at a particular time or noise.

5. REFERENCES

Z. Bien & J. X. Xu, *Iterative Learning control. Analysis, Design, Integration and Applications.* Kluwer Academic Publishers, 1998.

B.G. Dijkstra, O.H. Bosgra, Extrapolation of Optimal Lifted System ILC Solution, with Application to a Waferstage. *American Control Conference* (June, 2002).

B.G. Dijkstra, O.H. Bosgra, Noise suppression in buffer-state Iterative Learning Control, applied to a high precision wafer stage. *Conference on Control Applications*, Glasgow (September, 2002).

L. Cohen, *Time Frequency analysis*, Prentice Hall, NJ, 1995.

W. S. Levine (Ed.), *The control handbook*, 1996 by CRC Press, Inc.

K.L. Moore, *Iterative Learning contrrol for deterministic systems*, Springer Verlag, London 1993.

M. Steinbuch, Repetitive control for systems with uncertain period-time, *Automatica*, vol. 38(42), 2002.

M. Tomizuka, Zero Phase Error Tracking Algorithm for Digital Control. *Journal of Dynamic Systems, Measurement and Control*, 109, 65-68, 1987.

R. Tousain and E. Van der Meché, Design strategies for iterative learning control based on optimal control. *Conference on Decision and Control*, December 2001.

M. van de Wal, G. van Baars, F. Sperling, O. Bosgra, Multivariable $\mathcal{H}_\infty/\mu$ feedback control design for high-precison wafer stage motion. *Control Engineering Practice* 10 (2002) 739-755.