Adaptive iterative learning control for high precision motion systems

Adaptive Iterative Learning Control for High Precision Motion Systems

Iuliana Rotariu, Maarten Steinbuch, *Senior Member, IEEE*, and Rogier Ellenbroek

***Abstract*—Iterative learning control (ILC) is a very effective technique to reduce systematic errors that occur in systems that repetitively perform the same motion or operation. However, several characteristics have prevented standard ILC from being widely used for high precision motion systems. Most importantly, the learned feedforward signal depends on the motion profile (setpoint trajectory) and if this is altered, the learning process has to be repeated. Secondly, ILC amplifies non-repetitive disturbances and noise. Finally, its performance may be limited due to position-dependent dynamics. This paper presents the design and implementation of a time–frequency adaptive ILC that is applicable for motion systems which executes the same motion or operation. It employs the same control system block diagram as standard ILC, but instead of a fixed robustness filter it uses a time-varying filter. By using the results of the time–frequency adaptive ILC, i.e., the shape of the learned feedforward signal, a "piecewise ILC" is proposed that leads to the design of a single learned feedforward signal suitable for different setpoints. The results are experimentally shown to work for a high precision motion system.**

***Index Terms*—Adaptive filtering, learning control systems, motion control, time–frequency analysis, time-varying filters.**

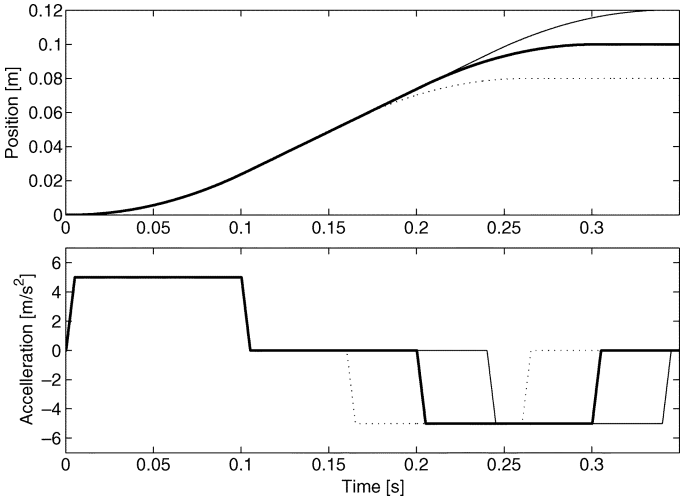

Fig. 1. Effects of extending or contracting the third-order setpoint (top) for larger or smaller die sizes on its acceleration profile (bottom). Since the considered system behaves largely as a rigid body, a suitable feedforward signal is very similar to this profile.

I. Introduction

THE ESSENTIAL steps in the manufacturing process of integrated circuits (IC's) are performed by lithographic machines called *wafer scanners*. An important module of these machines is the *wafer stage*. This is a high precision, 6 degrees-of-freedom (DOF) motion system, which repeatedly positions the silicon wafer with respect to the illumination optics. As integrated circuits (ICs) become smaller, the required precision increases proportionally. Positioning tolerances are currently in the order of nanometers, and can only be met using a very careful design of the machine's mechanics, actuators, electronics, software, measurement and control systems, etc.

In particular in control system design, well known concepts like proportional–differential (PD) feedback controllers including notch-filters and rigid body acceleration feedforwards are being used. In the future, more advanced designs will be needed to achieve the desired tracking performance [1], including multivariable feedback control.

Manuscript received July 27, 2005; revised January 27, 2007. Manuscript received in final form June 6, 2007. First published June 20, 2008; last published July 30, 2008 (projected). Recommended by Associate Editor I. G. Rosen.

I. Rotariu was with the Philips Applied Technologies, 5656 AE Eindhoven, The Netherlands. She is now with the Mapper Lithography, 2628 XK Delft, The Netherlands (e-mail: iuliana.rotariu@mapperlithography.com).

M. Steinbuch is with the Department of Mechanical Engineering, Control Systems Technology Group, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands (e-mail: m.steinbuch@tue.nl).

R. Ellenbroek was with the Philips Applied Technologies, 5656 AE Eindhoven, The Netherlands. He is now with the Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: r.m.l.ellenbroek@dcsc.tudelft.nl).

Digital Object Identifier 10.1109/TCST.2007.906319

For motion systems that repeatedly perform the same movement *iterative learning control* (ILC) [2] can be applied. This technique uses an iterative process to incorporate past control information, i.e., the tracking error signal of the previous iteration (called trial), into the construction of a new feedforward signal. ILC can be viewed as a trial-domain integral action with zero asymptotic error for constant disturbances. Such disturbances in the trial domain are hence equivalent to fixed reference or fixed disturbance profiles. Although ILC leads to good tracking performance [3]–[5] several drawbacks remain. First, the learned signal depends on the setpoint, so the whole learning process has to be repeated if this setpoint is altered. For motion systems and wafer scanners in particular, this constitutes the main drawback, since the setpoint is related to the frequently changing size of the die to illuminate (see Fig. 1). In Fig. 1, a typical third-order motion setpoint is shown, for three different scan lengths. For another die, hence another setpoint length, the learning process has to be repeated (typically 5–10 additional scans) and this will lower the overall wafer throughput of the machine. Second, ILC amplifies noise and other non-repetitive disturbances [6], [7]. Finally, its performance is limited due to position-dependent dynamics when the position of the setpoint varies.

This paper proposes a solution for reducing the dependency of the learned feedforward signal on the reference setpoint. In [8] and [9], we show that the proposed method does not lead to noise amplification. The final problem, position-dependent behavior, is addressed in [10] and [11].

Standard ILC—see for an extensive overview, and application to a wafer stepper system the work of [4]—consists of the design of a learning filter L and a robustness filter Q (see Fig. 2). A compromise must be made in the design of the robustness filter, since this filter influences both the reduction of the repetitive

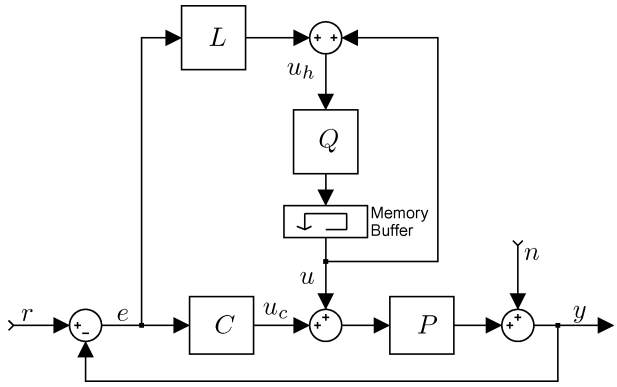

Fig. 2. Block-schematic for standard ILC.

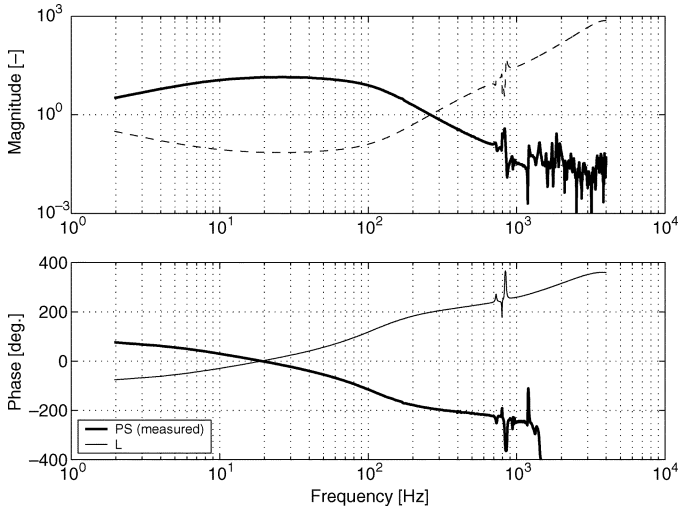

Fig. 3. Bode-plot of the (SISO) measured frequency response function $S_p(j\omega)$) at the center wafer position (thick line), together with the learning filter $L(j\omega)$ (thin line).

error and the amplification of noise. In this paper, a time-varying robustness filter is introduced that adapts according to the momentary frequency content of the feedforward signal. This leads to a time–frequency adaptive ILC similar to [12] and [13], while achieving an equivalent suppression of repetitive errors [9]. The main difference is that, in [12], the time–frequency analysis of feedback signals is used in order to design a feedforward signal that is updated iteratively based on the output of a B-splines network. In [13], by means of time–frequency analysis, one designs an adaptive nonlinear controller using a different ILC block diagram than the one we employ here. The control signals in our research exhibit a more pronounced multicomponent behavior and require more accurate signal processing by means of time–frequency analysis.

The adaptive ILC presented in this paper is based on the time–frequency analysis of the error signals. First of all, this analysis identifies the deterministic (stationary and non-stationary) and stochastic effects present in measured signals, and is used to design an adaptive cutoff frequency of the robustness filter Q. For the considered system, the cutoff frequency of the time-varying Q filter could be taken even zero for those time intervals where the measured control signals do not show any deterministic content. Based on this, the adaptive ILC leads to a single learned feedforward signal that can be used for different setpoints. The technique—called *piecewise ILC*—solves part of the setpoint dependency of ILC for classes of systems that show little position-dependent behavior.

This paper is organized as follows. In Section II standard ILC is applied to the motion system. Section III discusses joint time–frequency analysis techniques in general and its application to control signals relevant for ILC. The design of a time-varying robustness filter is then described in Section IV, for which a suitable bandwidth profile is sought in Section V. Finally, the concept and experimental results for the wafer scanner using piecewise ILC are shown and discussed in Section VI.

II. Iterative Learning Control

In this section, standard ILC is applied to the motion system. Consider a SISO LTI closed loop system with a plant $P(s)$ and a stabilizing feedback controller $C(s)$.

When the setpoint r is the only disturbance acting on the loop (i.e., $n = 0$), the servo error is $e = e^r = S(s)r$ with $S(s) = (1+P(s)C(s))^{-1}$ the sensitivity function. In the sequel, we will omit the addition of the Laplace symbol (s) for the signals, and we will assume zero initial conditions. Standard ILC [14] is an iterative technique that consists of the construction of a suitable feedforward signal u such that the servo error e is reduced. Its design consists of the design of the learning filter $L(s)$ and the robustness filter $Q(s)$ (Fig. 2). The ILC update law for trial k is given by

$$u_{k+1} = Q(s)\left(u_k + L(s)e_k\right) \tag{1}$$

$$e_k = e^r - S_p(s)u_k - n_k \tag{2}$$

where $S_p(s) = S(s)P(s)$, e_k is the error signal, u_k the feedforward signal, and n_k an output disturbance. Note that $Q(s)$ and $L(s)$, which modify the sequence $\{u_k\}_k$, are designed offline.

The *fixed-point theorem* [15], provides a sufficient condition for the convergence of ILC [14]

$$\left|Q(s)\left(1 - L(s)S_p(s)\right)\right| < 1 \quad \forall \quad s \in j\mathbb{R}. \tag{3}$$

From this criterion, it follows that the learning filter $L(s)$ should approximate the inverse of the process-sensitivity function $S_p(s)$, such that $L(s)S_p(s) \approx 1$. It also shows the role of the robustness filter $Q(s)$, which is used to make sure that the criterion (3) is satisfied for all frequencies where $|1 - L(s)S_p(s)| \not< 1$.

Note that convergence criterion (3) holds only when the robustness filter is steady state. Section IV will address the convergence criterion for time–frequency adaptive ILC, i.e., when the Q-filter is time-varying.

Some implementation issues will be briefly discussed. A first step when applying ILC is the design of a learning filter $L(s)$. Here, the ZPETC algorithm of Tomizuka [16] has been used to provide a stable approximation of the inverse of the modeled process-sensitivity function $S_p(s)$ (see Fig. 3). In Fig. 3, a measured frequency response function is shown of the scanning direction motion system of the wafer scanner under investigation. Some of the mechanical resonances inherent in such systems are clearly visible at higher frequencies. In the same figure the frequency response is plotted of the learning filter designed.

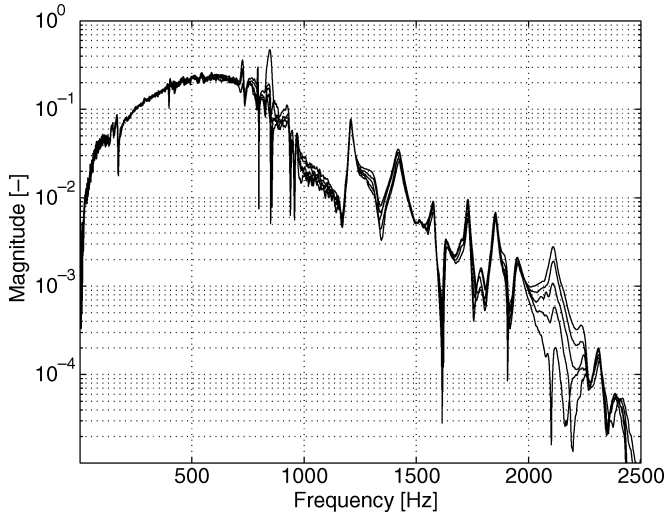

Fig. 4. Convergence criterion from (3) plotted for different positions along the setpoint. Differences between the plotted lines are due to position-dependent dynamics of the system, which become significant when applying ILC frequencies around 830 Hz.

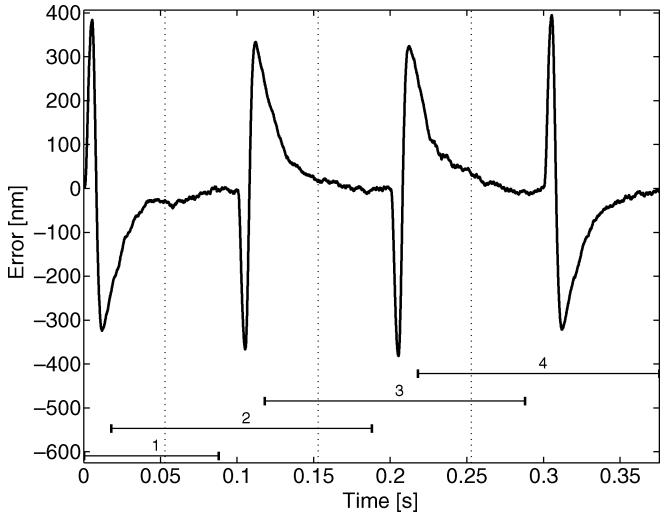

Fig. 5. Measured servo error, while performing a normal third-order setpoint with a rigid body acceleration feedforward. The numbered intervals are the partially overlapping pieces for time–frequency analysis. The analyzed pieces are sewn together at the dotted lines.

When applying standard ILC to the motion system for a given scanning trajectory, the cutoff frequency has to be tuned such that the learned servo error is convergent overall along the scanning trajectory, i.e., the fixed cutoff frequency is reduced such that one accounts for position dependent dynamics within the scanning trajectory [8]. The Q-filter is also used in order to increase the robustness of the ILC against high-frequent noise amplification and plant/model mismatch. Therefore, the Q-filter is given a low-pass characteristic. However, low-pass filters introduce a phase shift to the filtered signal, which results in a reduced tracking performance. Therefore, a filtering procedure is used that first filters a signal normally and successively restores its phase by applying the same filter backwards in time (MATLAB's `filtfilt` function).

A fourth-order Butterworth filter with a bandwidth of 500 Hz, implemented using this technique has been found to insure convergence for all positions along an arbitrary, fixed setpoint and a very good tracking performance, i.e., factor ten improvement with respect to tracking performance when acceleration feedforward control is applied, see [4]. Fig. 4 shows a magnitude plot of the convergence criterion (3), evaluated for various positions on the wafer.

When a standard third-order setpoint is performed with a rigid body mass feedforward, a typical servo error results, as shown in Fig. 5. The errors are obtained with a feedback bandwidth of about 100 Hz, and a standard acceleration feedforward. The residual error is in the order of 300 nm. Using ILC the results are as shown in Fig. 6, after five learning iterations only. Clearly, the error reduced significantly, up to a factor 10.

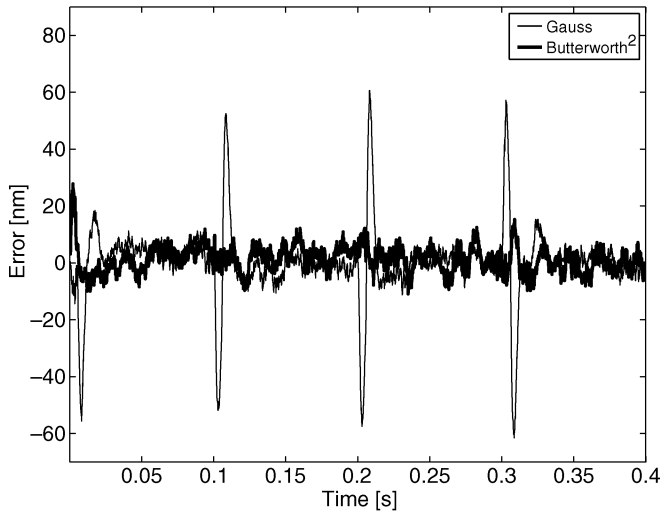

Fig. 6. Time domain error signals obtained after convergence of Standard ILC based on a fourth-order Butterworth filter (implemented using the `filtfilt` method) and on a Gaussian filter; both with a bandwidth of 500 [Hz].

III. JOINT TIME–FREQUENCY ANALYSIS

In this section, we shortly introduce the concept of quadratic time–frequency analysis of measured signals by means of Wigner distribution as a quadratic time–frequency representation [8], [17].

The integration of time–frequency analysis with the field of iterative learning control will be investigated for two purposes. First of all to identify the deterministic (stationary and nonstationary) and stochastic effects present in measured signals. Second, time–frequency analysis is performed on the feedforward signal u_h (see Fig. 2). This information will be used in Section V to find a suitable profile for the bandwidth of the time-varying robustness filter described in Section IV.

The Wigner distribution is defined by

$$W(t,f) = \frac{1}{2\pi} \int_{-\infty}^{\infty} s^* \left(t - \frac{\tau}{2}\right) s \left(t + \frac{\tau}{2}\right) e^{-2\pi j f \tau} d\tau \qquad (4)$$

where $t,\ f \in \mathbb{R}$, s^* is the complex conjugate of the analyzed time-signal s and f the frequency in hertz. The distribution is real-valued and can—due to its quadratic form—be physically interpreted as the distribution of the signal's energy over both

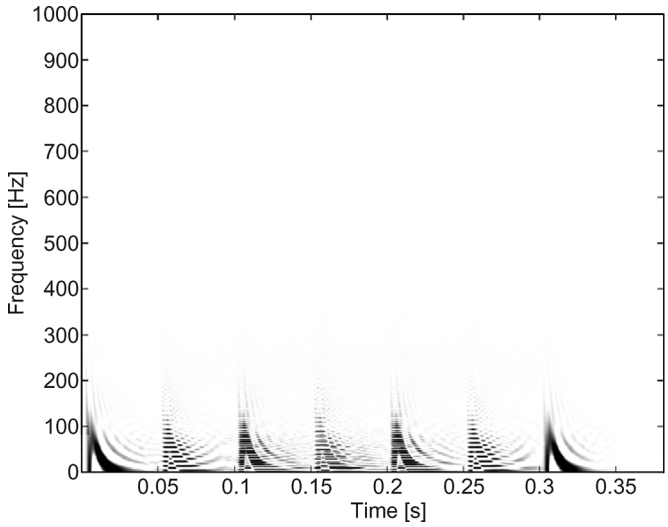

Fig. 7. Wigner distribution of the error signal in Fig. 5, created using [18]. Darker means higher relative energy. Seven intervals can be identified with significant energy content, of which three spurious ones due to cross-terms. Note that also the non-spurious third and fifth peaks are visibly distorted by cross-terms.

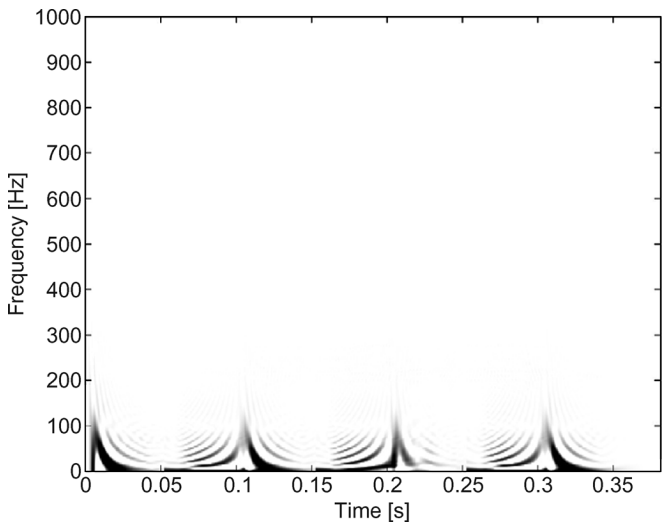

Fig. 8. Piece-wise Wigner analysis of the error signal in Fig. 5. The cross-terms are significantly reduced in comparison to Fig. 7.

time and frequency. Although the Wigner Distribution is especially appropriate for the analysis of nonstationary multicomponent signals [17], the main deficiency in the time–frequency analysis based on Wigner distribution is the cross-term inference: each pair of signal components or signal component and noise create one additional cross-term in the spectrum, thus the desired time–frequency representation may be confusing.

For the purpose of this research, we propose and explain next a relatively easy numerical method that eliminates these inference terms, i.e., piecewise clipping algorithm of the considered nonstationary multicomponent signal.

As mentioned before, for a standard third-order setpoint with a rigid body mass feedforward, a typical servo error results, as shown in Fig. 5. This signal shows highly repetitive behavior and contains short intervals with high energy, corresponding to intervals of non-zero jerk in the setpoint. The Wigner distribution of this signal will contain many cross-terms that are situated—as explained—between all auto-terms, which results in the image depicted in Fig. 7. Instead of the four expected high energy peaks, as many as seven can be identified.

However, the cross-terms that dominate this image are eliminated if the signal is cut into four pieces, such that each piece contains only one peak (see Fig. 5). Although this comes at the cost of frequency resolution, it is easily justified by the advantage of cross-term reduction and reduced computational costs. Note from Fig. 5 that the pieces are chosen partially overlapping. This keeps the partial time-signals as long as possible and thus restricts the loss of frequency resolution.

Piece-wise analysis of the servo error signal in Fig. 5 results in the image shown in Fig. 8. This shows significantly less cross-terms when compared to the one in Fig. 7. The same piecewise approach has been used to analyze the feedforward signal u_h (see Fig. 2), with a similar effect on cross-terms.

IV. ILC With Adaptive Robustness Filter

When the time–frequency analysis (see Fig. 8) is applied for the error signal from Fig. 5, it can be observed that the energy of deterministic, high frequencies in the error signal is concentrated around the moments of non-zero jerk in the setpoint. Therefore, a high Q-filter bandwidth is not required everywhere in time.

Next, we introduce the time–frequency adaptive Q-filter in the context of ILC. With the same learning filter L as for standard ILC, we replace the fixed Q-filter with a time-varying Q-filter $Q_k(s, \overline{t}, \Omega_k(\overline{t}))$, namely a zero-phase Butterworth filter of order n and cutoff frequency $\Omega_k(\overline{t})$, where $\overline{t} \in [t_{0(k)}, t_{0(k)} + T]$, $t_{0(k)}$ is the initial time of the kth cycle, T is the time required to perform the trajectory. The cutoff frequency $\Omega_k = \Omega_k(\overline{t})$ may vary throughout the length of each trial. Therefore, at each time instant $\overline{t}$, the Q-filter might change its cutoff frequency. In what follows, we denote by $\Gamma_k(t, \overline{t}, \Omega_k(\overline{t}))$ the inverse Fourier transform of the Butterworth filter $Q_k(s, \overline{t}, \Gamma_k(\overline{t}))$ as a function in the variable s

$$Q_k\left(s, \overline{t}, \Omega_k(\overline{t})\right) \xrightarrow{\mathcal{F}^{-1}} \Gamma_k\left(t, \overline{t}, \Omega_k(\overline{t})\right). \tag{5}$$

Convergence Criterion of the Adaptive ILC: In [19] and [20] a rigorous convergence analysis for a similar adaptive ILC has been carried out. Next, we prove briefly the convergence of the applied adaptive algorithm by use of an uniform version of the fixed point theorem [21]. As explained in [8], in the case when the adaptive filter Q is used, the update formula (2) remains the same while (1) becomes

$$u_{k+1}(\overline{t}) = \int_{-\infty}^{\infty} \Gamma_k\left(\tau, \overline{t}, \sigma_k(\overline{t})\right)(u_k + Le_k)(\overline{t} - \tau)d\tau \tag{6}$$

known as the nonstationary convolutional integral [22], [23] which is an extension of the convolutional method to nonstationary processes. This theory can apply to any linear, nonstationary filter, with arbitrary time and frequency variation, in

time, Fourier or mixed domain. In [23], it is also shown that this theory applies to general time-variant filtering and forward and inverse Q-filtering. Based on (2), (6), and the uniform fixed point theorem, a sufficient but not necessary condition for the convergence of the adaptive ILC can be derived. This convergence criterion can be written as

$$\left|Q_k\left(s,\overline{t},\Omega_k(\overline{t})\right)\left(1-L(s)S_p(s)\right|<1-\varepsilon \qquad (7)$$

$\forall\, s\in j\mathbb{R}, \forall \overline{t}\in[t_{0(k)},t_{0(k)}+T], \forall k\in\mathbb{N}$. where $\varepsilon>0$ is an arbitrary chosen fixed small parameter. From this criterion, it follows that the learning filter L should approximate the inverse of the process-sensitivity function S_p such that $L(s)S_p(s)\approx 1$. The role of the adaptive robustness filter Q is to make sure that the criterion (7) is satisfied for all frequencies where $|1-L(s)S_p(s)|$ is not very small (from Fig. 4 it can be seen that this is valid up to 830 Hz). The Q-filter is therefore designed to have a varying bandwidth of not more than about 500 Hz and a fast roll-off, such that the convergence criterion is always satisfied for all frequencies and any time instance in an uniform way with respect to the iteration index k and the time $\overline{t}$. It follows that

$$\sup_{k,\overline{t}}\left|Q_k\left(s,\overline{t},\Omega_k(\overline{t})\right)\left(1-L(s)S_p(s)\right|<1 \quad \forall\, s\in j\mathbb{R}. \qquad (8)$$

When designing the adaptive filter Q, it is easy to take care numerically that the convergence criterion (8) is satisfied.

We shall apply the adaptation mechanism along the whole trajectory length time-interval. First, the design of the time–frequency Q-filter comprehends the use of the Wigner distribution. Using Wigner distribution we identify the high and low frequency components of the control signals (namely, the signals u_h at each iteration) as well as their energy levels so that one could determine if there is deterministic (low or high frequency) system dynamics at a particular time instant (for a given set-point, time is equivalent to position) or just measurement noise. The stochastic definition of the Wigner distribution allows us to identify the *stochastic effects* presented in the control signals (implicitly in the servo errors) and to consider these effects in the control strategy.

The Q-filter bandwidth (cutoff frequency of the Butterworth type filter Ω_k) varies according to the current frequency contents present in the system signals. The filter effectively changes its bandwidth (cutoff frequency) as a function of time and the *high frequency dynamics will enter into the learning feedforward controlled process at the appropriate time instances*. Also, the bandwidth of the filter is reduced when the system dynamics do not exhibit high frequency components and, therefore, it *avoids noise amplification when applying learning feedforward control*. Therefore, the adaptation algorithm will adjust the bandwidth frequency as a function of time to maximize the tracking performance while still maintaining a good noise performance.

Remark 1: The time–frequency adaptive Q-filter can be seen as a time-varying low-pass filter. The filter might change very fast (the motion control considered has a sampling rate of 0.125 ms). The fast switching between the cutoff frequency of different Q-filters which correspond to different time instances is a major issue in switching control and hybrid system area. For stability results of switched systems, which can indicate a way to handle with the Q-filter, we refer to [24]. In the case of the considered motion system, the cutoff frequencies are changing smoothly enough from one time instant to another. The switching control law does not affect the stability of the system.

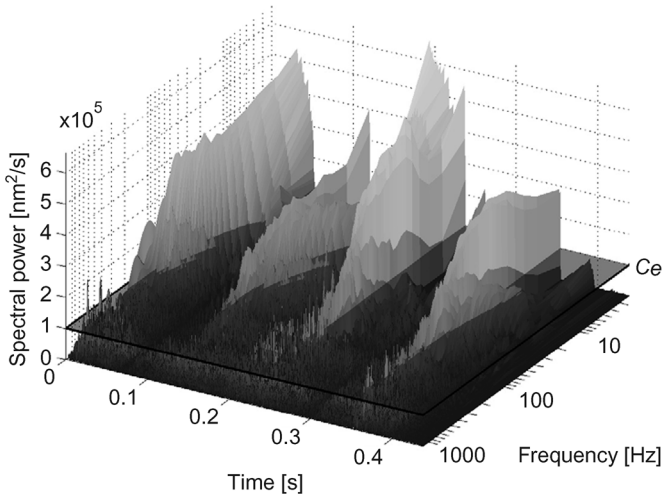

Fig. 9. 3-D image of the Wigner distribution of the error signal filtered by the learning filter. The semi-transparent horizontal plane represents the energy value C_e that discriminates deterministic signal components from noise.

Remark 2: The algorithm explained in this section is adaptive not only over the time interval within a considered trial, but over iterations also. In this paper, we show the time domain performance results when implementing the same time–frequency adaptive filter for all trials. Therefore, the bandwidth of the Q-filter changes in time within one trial, but does not adapt its profile from one iteration to another one. Adaptive tuning of the Q-filter bandwidth profile over iterations will be shown in a future paper.

V. Design of a Bandwidth Profile

In this section, a bandwidth profile is constructed the time-varying robustness filter described in Section IV. Information obtained through time–frequency analysis (see Sections III and IV) is used to update the bandwidth profile of the Q-filter during the ILC process. This robustness filter is going to be implemented in the closed loop LC configuration (see Fig. 2).

Consider the time vector $\overline{t}=(t_i)_{i\in\mathbb{N}}$ and the initial bandwidth profile $\Omega_0(\overline{t})$. The elements of the vector $\Omega_0(\overline{t})$ are chosen equal small values, whereas the initial feedforward signal u is chosen zero.

A step-by-step description of the adaptive learning process will now be given. Note that these steps are taken with *each* iteration.

Step 1) The learned feedforward signal is implemented, such that a corresponding error signal can be measured. This signal is filtered by the learning filter, with which the feedforward signal from the previous iteration is summed to obtain u_h (see Fig. 2). The piecewise Wigner analysis described in Section III will be performed on this signal. This provides information on the location of relevant signal components in this signal, which is to be filtered by the time-varying robustness filter from Section IV.

Step 2) Observe in Fig. 9—which shows the Wigner distribution from step one as a 3-D surface—that the energy

of noise on u_h is significantly smaller than that of the deterministic disturbances. Based on a time–frequency analysis of the noise measured in the system in rest (i.e., at stand still), an energy level C_e represented by the horizontal plane in Fig. 9 is chosen that discriminates between noise and deterministic signal components at every point in the time–frequency plane.

At a given arbitrary iteration k, a frequency envelope $F_{max,k}(\bar{t})$ is now constructed, that encompasses the frequencies of all signal components at each time-instant, whose energy exceeds C_e

$$F_{\max,k}(\bar{t}) = \max\left(\omega_k(\bar{t})\right), \text{ for } W_{u_{h,k}}\left(\bar{t}, \omega_k(\bar{t})\right) \geq C_e \quad (9)$$

where $W_{u_{h,k}}$ is the Wigner distribution of the error signal u_h at iteration k. Note that $\omega_k(\bar{t})$ and $F_{\max,k}(\bar{t})$ are not functions in the mathematical sense.

Step 3) The envelope $F_{\max}(\bar{t})$ is used as a gain in an adaptive update law. This law changes the bandwidth profile $\Omega(\bar{t})$ after each iteration, when the effects of the previous change on the measured error are evaluated. After a new bandwidth profile has been implemented, its benefit is evaluated by the function $\Delta N_k(\bar{t})$, which compares the local ℓ_2 norm of the current error to that of the error at the previous iteration, such that

$$\Delta N_k(\bar{t}) = N_k(\bar{t}) - N_{k-1}(\bar{t}) \quad (10)$$

where

$$N_k(t_i) = \sum_{j=i-T_w/2}^{i+T_w/2} e_k^2(t_j) \quad (11)$$

$T_w > 0$ gives the width of the window where the signals are locally compared.

The difference $\Delta N_k(\bar{t})$ becomes a second gain in the update rule, which is given by

$$\begin{aligned} \Omega_{k+1}(\bar{t}) &= \Omega_k(\bar{t}) + \Delta\Omega_k(\bar{t}) \\ \Delta\Omega_k(\bar{t}) &= F_{\max,k}(\bar{t}) \cdot \Delta N_k(\bar{t}) \cdot G_u \cdot K_k(\bar{t}) \end{aligned} \quad (12)$$

where G_u is a global gain to be tuned later, and

$$K_k(\bar{t}) = -\text{sign}\left(\Delta\Omega_{k-1}(\bar{t})\right) \quad (13)$$

is introduced to add the following logic to the mechanism: if the bandwidth was previously increased ($\Delta\Omega_{k-1}(t_i) > 0$), while the error decreased ($\Delta N_k(t_i) < 0$), this change was beneficial and the bandwidth may be further increased. On the other hand, if an increase in the bandwidth resulted in a larger error, this was obviously not the case and the bandwidth should be lowered again. The combination $\Delta N_k(\bar{t}) \cdot K_k(\bar{t})$ results in this kind of update behavior.

In order to minimize the loss of tracking performance due to bandwidth variation (see *Remark 1*, Section IV), the obtained bandwidth profile is smoothed before it is used. This is done by applying a low-pass Gaussian filter with a bandwidth of approximately 20 Hz. Simulations have shown this bandwidth value to give the best results.

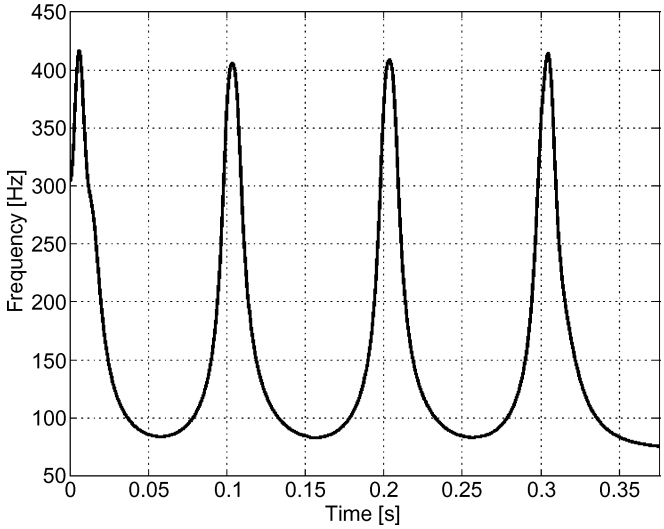

Fig. 10. Approximately converged bandwidth profile $\Omega_k(\bar{t})$ for $k = 30$ resulting from the adaptive algorithm.

Finally, the feedforward signal $u_{f,k+1}$ for the next trial is obtained by performing the time-varying filtering operation on $u_{h,k+1}$ (see Step 1).

Tuning and Results: In the description of the adaptive algorithm, three parameters have been introduced that have been designed using simulations. They are the energy level $C_e > 0$, the global update gain $G_u > 0$, and the window width $T_w > 0$.

C_e should be chosen such that the level plane in Fig. 9 lies just above the highest peaks of the Wigner distribution of the noise (determined at stand still). However, it cannot be chosen too tight above the real noise level as the Wigner distribution of u_h will contain cross-terms between noise and deterministic signal components, which could exceed the plane and thus affect the envelope $F_{\max}(\bar{t})$.

The bandwidth update gain G_u influences the highest bandwidth values of $\Omega_k(\bar{t})\,\forall\, k \in \mathbb{N}$, see (12). In (11) and (12), as well as in simulations, it has been observed that the window width T_w influences the width of the high-bandwidth peaks around the jerk-moments (see Fig. 10). It has been tuned such that the penalty on tracking performance (see Section IV) plays no significant role.

Evaluation: For the considered test-rig, the time–frequency adaptive ILC achieves a tracking performance that is comparable to standard ILC [8], while the bandwidth of its Q-filter is much lower for most time-instances. Therefore, it leads to a smaller amplification of non-repetitive disturbances and noise.

After tuning, a bandwidth profile is typically obtained as depicted in Fig. 10. A learned feedforward signal is produced, which looks very much as expected (see Fig. 11). It shows high-frequent behavior around time-instants of non-zero jerk in the setpoint, while it is almost constant in between these intervals, where the Q-filter has a low bandwidth.

Remark: When the bandwidth profile is changed by the update rule (12), this affects the tracking error in three ways. It changes the suppression of repetitive disturbances (I) as well

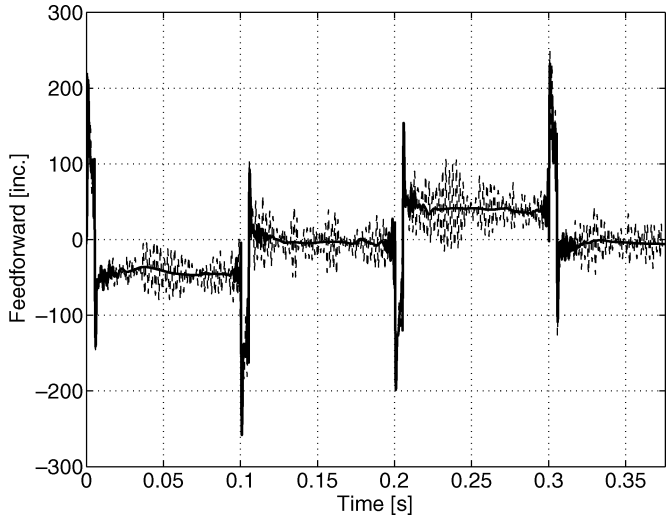

Fig. 11. Comparison of feedforward signals resulting from standard (dashed line) and adaptive ILC (thick line). Note that in order to improve the axis scales, the signals are plotted as the difference to the rigid body acceleration feedforward.

as the amplification of noise (II). Further, it leads to a different performance penalty due to bandwidth variation (III). Since a change in noise amplification has a much smaller influence on the evaluation function $\Delta N_k(\bar{t})$ than the other two effects, this has relatively little influence on the realization of the bandwidth profile. Therefore, the bandwidth profile found using the adaptive algorithm does not necessarily lead to the smallest tracking error achievable through application of ILC with a time-varying robustness filter. The results are very much dependent on the values of the parameters G_u, C_e, T_w and on the bandwidth smoothing operation, which complicates the tuning process.

VI. Piecewise ILC

It has been shown in the previous section, that learning is only important around time-instants of non-zero jerk in the setpoint where unwanted excitation of the system dynamics must be suppressed. Sufficient tracking performance is obtained by applying a constant feedforward signal in between these intervals.

Based on this knowledge, the concept of *piecewise ILC* will now be introduced. From the feedforward signal given in Fig. 11, which has been learned for a particular setpoint, a feedforward signal has been derived that is suitable for different setpoints.

As explained in the introduction, the length of the illumination interval is the most relevant parameter of the setpoint to be able to vary. In Fig. 12, the four jerk intervals of a third-order setpoint are numbered and shown in the acceleration profile. When a die of different size needs to be illuminated, the learned feedforward signal is split into an acceleration part (pieces 1 and 2) and a deceleration part (pieces 3 and 4). For larger dies, the signal is extended by inserting zeros in between the two parts, and for smaller dies both parts can be superpositioned. The latter can be justified because the system shows highly linear behavior.

Remark: Although the piecewise approach has only been used to vary the illumination length, more elaborate variations

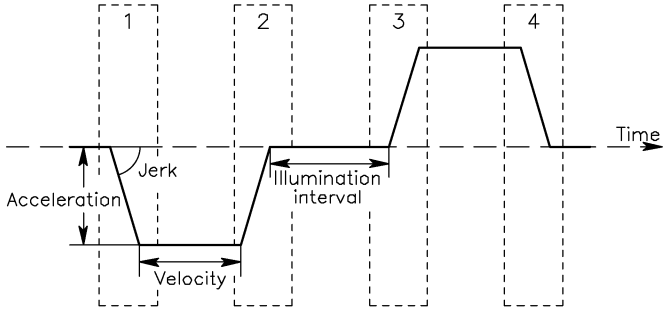

Fig. 12. Piece-wise ILC uses the four marked pieces of the learned feedforward to construct a suitable feedforward for many third-order setpoints.

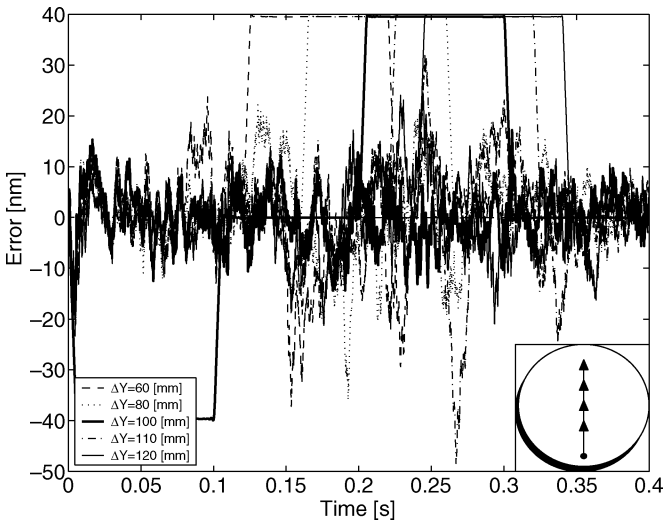

Fig. 13. Measured error signals for application of piecewise ILC for various step sizes, but starting at the same position (see wafer-position schematic at the bottom-right). The signals are plotted together with their corresponding acceleration profiles (not to scale).

should be possible for third-order setpoints. The velocity during the illumination interval may be changed by adjusting the distance between pieces 1 and 2 as well as between 3 and 4, while the acceleration levels may be adjusted by a linear scaling operation on the whole signal. The only constraint is that the jerk level is related to the acceleration level.

Experimental Results: When the explained methods concerning the adaptation of the scan length are implemented on the wafer scanner test rig, very good results are obtained (see also [9]). Fig. 13 shows the servo errors obtained for both longer and shorter setpoints. The error signals obtained for different illumination lengths show peaks that do not occur for the learned setpoint and are—as will be discussed in the next paragraph—due to position-dependent disturbances.

VII. Conclusion

In this paper, time–frequency analysis of the servo error of a wafer stage motion system has led to the design of a time-varying robustness filter for ILC. By varying the bandwidth of a zero-phase Butterworth filter over time, such that only deterministic disturbances are learned, the amplification of noise can be restricted. For the considered system, the cutoff frequency of the time-varying Q-filter should be taken as small as possible, i.e., ideally constant zero values for some time intervals

where the measured control signals do not show any deterministic content. Several issues concerning time-varying filtering that affect the learning performance have been identified. To insure a good tracking performance, a bandwidth profile must be very smooth. Further, the frequency domain characteristics as well as the shape of the corresponding impulse response functions of the filter at each time-instant are important. Based on the time–frequency analysis of the learned feedforward signal we proposed "piecewise ILC" that leads to design a single learned feedforward signal suitable for different setpoints.

Time-frequency adaptive ILC applied for a fixed setpoint improves the tracking performance for all positions along the trajectory, i.e., a factor ten improvement with respect to tracking performance when acceleration feedforward is applied. The comparison in [9] shows that the proposed method achieves a similar tracking performance as standard ILC, that it converges faster, and that it reduces noise amplification.

Further research might focus on the combination of wavelet analysis and filtering [25], which might provide a solution to both the cross-term problem in time–frequency analysis and the performance loss due to time-varying filtering, see also [7], [26].

Other applications of time-varying filtering in the field of ILC should also be explored. For systems with significant position-dependency and thus time-varying dynamics, the benefit of a time-varying robustness filter looks interesting. However, non-stationary behavior is not necessarily restricted to the robustness filter of ILC: a time or position-dependent learning filter is also a possibility. One promising way is the use of the lifted ILC approach, see for instance [5], [27]. The work reported here can also be used for design of the bandwidth profile for such a method.

REFERENCES

[1] M. van de Wal, G. van Baars, and F. Sperling, "Multivariable H-infinite feedback control design for high precision wafer stage motion," *Control Eng. Practice, IFAC*, vol. 10, pp. 739–755, 2002.

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

[3] I. Rotariu, B. Dijkstra, and M. Steinbuch, "Standard and lifted approaches of iterative learning control applied on a motion system," in *Proc. 16th Int. Symp. Math. Theory Netw. Syst.*, 2004, p. 6.

[4] D. de Roover, "Motion control of a wafer stage, a design approach for speeding up IC production," Ph.D. dissertation, Dept. Mech. Eng., TU Delft, NL, Delft, The Netherlands, 1997.

[5] B. Dijkstra, "Iterative Teaming Control," Ph.D. dissertation, Dept. 3mE, DCSC, TU Delft, NL, Delft, The Netherlands, 2004.

[6] E. Rogers, K. Galkowski, D. H. Owens, T. Al-Towlem, J. D. Ratcliffe, and P. Lewin, "2D linear control systems—From theory to experiment to theory," in *Proc. 15th Int. Symp. Math. Theory Netw. Syst.*, 2002, p. 6.

[7] R. Merry, M. van de Molengraft, and M. Steinbuch, "The influence of disturbances in iterative learning control," in *Proc. IEEE Int. Conf. Control App.*, 2005, vol. 5, no. 9, pp. 974–979.

[8] I. Rotariu, R. Ellenbrock, and M. Steinbuch, "Time-frequency analysis of a motion system with learning control," in *Proc. American Control Conf., Denver Amer. Control Conf.*, 2003, pp. 3650–3654.

[9] I. Rotariu, R. Ellenbrock, G. van Baars, and M. Steinbuch, "Scan-length independent iterative learning control applied to a wafer stage motion system," in *Proc. Eur. Control Conf.*, 2003, p. 6.

[10] B. Hennen, I. Rotariu, and M. Steinbuch, "Time-frequency analysis of position-dependent dynamics in an iteratively controlled waferstage," in *Proc. American Control Conf.*, New York, 2007, pp. 570–575.

[11] K. van Berkel, I. Rotariu, and M. Steinbuch, "Cogging compensating piecewise iterative learning control for variable setpoints with application to a wafer stage," in *Proc. Amer. Control Conf.*, New York, 2007, pp. 1275–1280.

[12] Y. Chen and K. Moore, "Frequency domain adaptive learning feedforward control," in *Proc. IEEE Int. Symp. Comp. Intell. Robot. Autom.*, 2001, pp. 396–401.

[13] D. Zheng and A. Alleyne, "Adaptive iterative learning control for systems with non-smooth non-linearities," presented at the ASME Int. Mechan. Eng. Congress Expo., New York, Nov. 2001, Paper DSC-24576.

[14] K. L. Moore, *Iterative Learning Control for Deterministic Systems*, ser. Adv. in Ind. Control. New York: Springer-Verlag, 1993.

[15] H. Brezis, *Analyse Functionelle*, 4th ed. Paris, France: Masson, 1993.

[16] M. Tomizuka, "Zero phase error tracking algorithm for digital control," *J. Dynamic Systems, Meas. Control*, vol. 109, pp. 65–68, 1987.

[17] K. Groöchenig, *Foundations of Time-Frequency Analysis*. Boston, MA: Birkhäuser, 2001.

[18] F. Auger, P. Flandrin, P. Concalves, and O. Lemoine, "Time-frequency toolbox for use with MATLAB," CNRS/Rice Univ., 1995–1996.

[19] D. Zheng and A. Alleyne, "Stability of a novel iterative learning control scheme with adaptive filtering," in *Proc. American Control Conf.*, Denver, CO, 2003, pp. 4512–4517.

[20] M. Tharayil and A. Alleyne, "A time-varying iterative learning control scheme," in *Proc. American Control Conf.*, Boston, MA, 2004, pp. 3782–3787.

[21] S. Badaev, S. Goncharov, and A. Sorbi, "Completeness and universality of arithmetical numberings," in *Computability and Models*. New York: Springer, 2002.

[22] G. F. Margrave, "Theory of nonstationary linear filtering in the fourier domain with applications to time-variant filtering," *Geophys.*, vol. 63, no. 1, pp. 244–259, 1998.

[23] T. Scheuer and D. Oldenburg, "Aspects of time-variant filtering," *Geophys.*, vol. 53, pp. 1399–1409, 1988.

[24] D. Liberzon and A. Morse, "Basic problems in stability and design of switched systems," in *Proc. IEEE Conf. Syst. Mag.*, 1999, vol. 19, pp. 59–70.

[25] T. A. Ridsdill-Smith, "Wavelet design of time-varying filters," in *Proc. 5th Int. Symp. Signal Proc. App. (ISSPA)*, 1999, vol. 2, pp. 599–602.

[26] R. Merry, M. van de Molengraft, and M. Steinbuch, "Iterative learning control with wavelet filtering," *Int. J. Robust Nonlinear Control*, vol. 18, no. 10, pp. 1052–1071, 2007.

[27] J. van de Wijdeven and O. Bosgra, "Residual vibration suppression using hankel iterative learning control," *Int. J. Robust Nonlinear Control*, vol. 18, no. 10, pp. 1034–1051, 2007.