15th Triennial World Congress, Barcelona, Spain

ADAPTIVE INVERSE-LATTICE LEARNING CONTROL

George Nikolakopoulos, Anthony Tzes [1]

University of Patras
Electrical and Computer Engineering Department
Rio GR-26500, GREECE

Abstract: In this article the design framework for an Adaptive Inverse Lattice Controller (AILC) with learning attributes, applicable to linear Auto Regressive (AR) systems, is presented. The utilized controller structure relies on the principle of Inverse Model Control (IMC) and its topology resembles that of a lattice filter. The adaptation rules depend on the identified system dynamics through an adaptive lattice filter. The identification scheme is extended with a proposed algorithm for the model order selection. Within the employed IMC–structure, an inverse lattice controller is utilized in the forward path in cascade with a lowpass detuning filter. As time progresses, the lattice filter estimates more accurately the system dynamics, and the learning scheme adjusts accordingly the attributes of the detuning filter. Simulation studies are used to investigate the efficacy of the suggested scheme. *Copyright © 2002 IFAC*

Keywords: Adaptive Lattice Filtering, Internal Model Control, Learning Control.

1. INTRODUCTION

The adaptive control problem for discrete jump systems (with multiple models) has received significant attention over the last years (M. Branicky, 1998; Narenda and Balakrishnan, 1997; Sun and Zheng, 2001). The suggested adaptive controllers account for the switching between the candidate systems and enhance the system's performance while maintaining stability. A typical adaptive controller is composed of the controller portion and the identifier module.

In an IMC structure, there is a natural coupling between the identification scheme and the control part (Morari and Zafiriou, 1989; Datta and L.Xing, 1998). This coupling affects the robustness and the overall system performance due to the system's variations. The IM-controller minimizes a cost which is a weighted function of the system's sensitivity function and its multiplicative uncertainty. If the performance attributes of the system's transient response are to be considered in the controller design process a learning scheme can be employed. Motivated by the "progressive learning" control design approach (Anderson and Kosut, 1991; Lee *et al.*, 1993), the uncertainty associated with the system dynamics is learned progressively through an identification scheme. This scheme employs a prefilter (Y. Zhu and P. van der Bosch, 2001; Rivera *et al.*, 1992), whose characteristics depend on the system's frequency spectrum and the adopted controller's objectives. Furthermore, the structure of the selected model in the identification process is that of a lattice filter (Friedlander, 1982).

The Adaptive Least Square Lattice Filtering (ALSL) (Haykin, 2001) is used for the identification of the system dynamics. The dynamics is mapped into the filter's structure through the values of the estimated reflection coefficients. In addition, the proposed extended ALSL–algorithm

[1] Corresponding author's coordinates:tzes@ee.upatras.gr, Tel: +30-(610)-996453

identifies the order of the system by monitoring the magnitude of those coefficients and the cumulative error-response. The proposed IMC controller corresponds to the inverse lattice filter, whose order varies. The controller is enhanced with a lowpass detuning filter with progressively increasing cutoff frequency, in order to provide balance between the system's robustness and its transient response performance.

This paper is structured in the following manner. In the next section, the formulation for the adaptive robust control design problem is stated. The proposed design embraces elements from the IMC–principle and the ALSL filtering technique. The enhancements related to the model order estimation, and the detuning filter are addressed in Section 3. The proposed approach is applied in simulation studies in the following section, while final remarks are offered in the last section.

2. ADAPTIVE ROBUST CONTROL DESIGN PROBLEM STATEMENT

Consider the AR linear discrete-time system with its dynamics expressed as

$$y(t) = G(z)u(t) + \nu(t) = \tilde{G}(z)(1 + l_m\Delta)u(t) + \nu(t)$$
$$= \left[\frac{b_d z^{-d}}{1 + \sum_{i=1}^{m} a_i z^{-i}}\right](1 + l_m\Delta)u(t) + \nu(t) \qquad (1)$$

where z^{-1} corresponds to the delay operator, $u(t)$, $(y(t))$ is the input (output) of the system, $\tilde{G}$ is the "nominal" plant description, l_m is the multiplicative uncertainty about the nominal plant, while the output measurements are corrupted by additive white noise $\nu(t)$. The system model is assumed to be stable and the delay term d is considered to be known *a priori*. The objective is to design an adaptive lattice controller relying on the IMC-principle to ensure the system's robust stability (Silva and Datta, 2001) in lieu of uncertainties in the parameters vector.

In this work, an identification scheme is coupled to a robust internal model controller, as shown in Figure 1. The Internal Model Controller $q(e^{j\omega})$ is generated by a cascade composition of: 1) an adaptive lattice controller $\tilde{q}(e^{j\omega})$, and 2) a low-pass filter $F(e^{j\omega})$ which detunes the controller characteristics at high frequencies in order to extend the system's robustness. The identification module relies on Recursive Least Squares Lattice (RLSL) filtering and estimates the plant dynamics by computing the corresponding reflection coefficients. The RLSL algorithm is extended with the proper criteria in order to identify the system order by monitoring: a) the magnitudes of the reflection coefficients, and b) the cumulative mean square error between the system and model response.

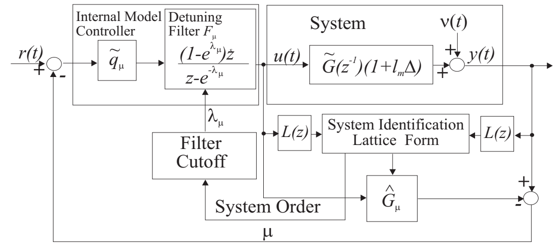

Fig. 1. Adaptive Inverse Lattice Control Structure

2.1 H_2–*Optimal Internal Model Controller Design*

For robust performance, the control objective is to minimize the infinity norm of the system's weighted sensitivity function $\epsilon(= 1 - Gq)$

$$\| w_p\epsilon \|_\infty = \sup_\omega |w_p\epsilon(j\omega)|$$

for all plants $\mathcal{G} = \left\{G : |(G - \tilde{G})\tilde{G}^{-1}| \le l_m\right\}$. The $|w_p|^{-1}$ represents an upper bound on the sensitivity function, since $|\epsilon(j\omega)| \le |w_p(j\omega)|^{-1}\ \forall\omega$ if and only if $\sup_\omega (|\tilde{\eta}l_m| + |w_p\tilde{\epsilon}|) \le 1$, where $\tilde{\eta}$ $(= 1 - \tilde{\epsilon} = \tilde{G}q)$ is the complementary sensitivity function for the nominal system $\tilde{G}$. The optimal controller design problem is formulated as

$$q = \arg\left\{\min_q \sup_\omega (|w_p\tilde{\epsilon}| + |\tilde{\eta}l_m|)\right\}, \qquad (2)$$

where q is the IMC feedback controller as shown in Figure 1. The philosophy behind the IMC–design consists of two steps, and although the resulting controller has no inherent optimality characteristics, it provides a good engineering approximation to the optimal solution of (2). The first step amounts to designing a controller $\tilde{q}$ for good nominal performance so that

$$\tilde{q} = \arg\left\{\min_{\tilde{q}} \| w_p\tilde{\epsilon} \|_2\right\} . \qquad (3)$$

In this case, the optimal sensitivity becomes $\bar{\epsilon} \triangleq 1 - \tilde{G}\tilde{q}$, and the optimal complementary sensitivity function $\bar{\eta} \triangleq \tilde{G}\tilde{q}$. Furthermore for the case, where the weight w_p reflects the particular input $(w_p = r(s))$, the cost function (3) to be minimized is the l_2 norm of the error $\| w_p\tilde{\epsilon} \|_2 = \| e_r \|_2$, and the resulting controller is H_2–optimal. The second step addresses the robust stability and performance issue. At high frequencies, when the multiplicative uncertainty l_m exceeds unity, $\bar{\eta}$ has to be rolled off. To achieve this action, $\tilde{q}$ is augmented (cascaded) by a low–pass filter F, as $q \triangleq \tilde{q}F$. The order of F is such that q is proper, and its roll–off frequency is selected so that the robust stability constraint $\| \tilde{\eta}l_m \|_\infty = \| \tilde{G}ql_m \|_\infty < 1$ is satisfied. The purpose of the filter F is to detune the controller, since it sacrifices performance for robustness. This is justified since the sensitivity

$\tilde{\epsilon} = 1 - \tilde{G}q = 1 - \tilde{G}\tilde{q}F$ (performance measure) is increased, while $\tilde{\eta} = \tilde{G}\tilde{q}F$ (robustness measure) decreases. Because of the $\tilde{G}$–system's inherent minimum phase characteristics, the optimal solution to the minimization of the cost in (3) is independent of the weight and equal to $\tilde{q} = \tilde{G}^{-1}$.

Since the nominal system $\tilde{G}$ is unknown, the estimated transfer function $\hat{G}$ will be used in the controller design process. Subsequently, the multiplicative error is $e_m = (\tilde{G} - \hat{G})\hat{G}^{-1}$, and the corresponding sensitivity function $\tilde{\epsilon} = (1 - \hat{\eta})(1 + e_m\hat{\eta})^{-1}$, where $\hat{\eta} = \tilde{G}\hat{G}^{-1}$. From Parseval's theorem the l_2 error–norm can be mapped into the frequency domain, where for the frequency range where $|e_m\hat{\eta}| \ll 1$ the control objective is bounded by

$$\| e_r \|_2 \leq \left[\frac{1}{\pi} \int_0^{\pi} |1 - \hat{\eta}|^2 |r|^2 d\omega \right]^{\frac{1}{2}} + \left[\frac{1}{\pi} \int_0^{\infty} |1 - \hat{\eta}|^2 |e_m\hat{\eta}|^2 |r|^2 d\omega \right]^{\frac{1}{2}} . \quad (4)$$

The effect of the identified transfer function to the cost objective is at the second term of (4), which includes the contribution of the multiplicative error. Therefore, $\hat{G}$ should be selected so as to minimize the control relevant identification cost:

$$\hat{G} = \arg\min_{\hat{G}} \left[\frac{1}{\pi} \int_0^{\pi} |1 - \hat{\eta}|^2 |(\tilde{G} - \hat{G})\hat{G}^{-1}\hat{\eta}|^2 |r|^2 d\omega \right]^{\frac{1}{2}} . (5)$$

2.2 *Data Prefiltering*

The system model, to be identified, is assumed to have the following linear regression structure

$$\hat{y}(t|\theta) = \hat{G}(z, \theta)u(t) + x_w(t) \ , \quad (6)$$

where θ is the system parameter vector (i.e., the lattice filter's reflection coefficients), and x_w corresponds to white noise. Let the filtered prediction error sequence be filtered through a stable linear filter $L(z)$:

$$\beta_F(t, \theta) = L(z)\beta(t, \theta) = L(z)\left[y(t) - \hat{y}(t|\theta)\right] . \quad (7)$$

The utilized L–filter can be used to enhance or suppress certain properties of the model, since it acts as a frequency weighting factor. The purpose of the identifier is to provide a parameter vector–θ that minimizes the following norm

$$\theta = \arg\left\{ \min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \left[\beta_F(t, \theta)\right]^2 \right\} \ , \quad (8)$$

where $\Phi_u(\omega)$ is the spectrum of the control input. Comparing this term to the integrand quantity in (5) infers the optimal prefilter L as

$$L(z) = \frac{(1 - \hat{\eta}(z)) \ \hat{G}^{-1}(z)\hat{\eta}(z)\left[r(z)\right]}{\mathcal{F}^{-1}\left(\Phi_u(\omega)\right)} \ ,$$

where $\mathcal{F}^{-1}$ corresponds to the inverse discrete Fourier transform. The computation of the optimal prefilter demands the knowledge of $\Phi_u(\omega)$ and $\hat{\eta}(z)$, which is rather difficult to be obtained *a priori*. In this research effort, the selection of the prefilter is based primarily to affect the convergence of the estimated transfer function to the actual one in the frequency domain. If $L(z^{-1})$ is selected as a lowpass filter, then the parameter vector–θ is computed in order to match the low-frequency spectrum of $\tilde{G}$ (from equation (8)). In our case, the cutoff of the lowpass filter is increased progressively as the estimator matches more closely the spectrum of the true system at the low–frequency end. In order to ascertain the efficiency of the suggested scheme in lieu of limited subsystem variations, the prefilter is selected to be identical to that of the detuning filter.

2.3 *System Identification via ALSL-Theory*

Upon computation of the $L(z)$ prefilter, the input and output data streams are filtered prior to their processing by the adopted identification scheme. The identifier, shown in Figure 2, rather than estimating the tapped weights in the direct transversal form of (1), computes in a recursive manner the $\kappa_{f,m}$–forward and $\kappa_{b,m}$–backward reflection coefficients characterizing the system's behavior.

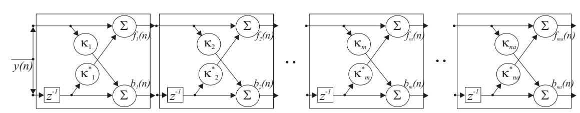

Fig. 2. Lattice Filter Structure

Under the assumption of an n_a–order model, its reflection coefficients are computed recursively based on the ALSL–algorithm (Haykin, 2001) (summarized in Table 1) using the *a posteriori* estimation errors.

The link between the direct and lattice form parameterizations can be found from the following relationships shown in Table 2 (for the AR–part with characteristic polynomial $1 + D(z) = 1 + \frac{D_{n_a}(z)}{z^{n_a}}$)

3. LEARNING ENHANCEMENTS TO THE ADAPTIVE IMC–DESIGN

The learning scheme, employed in this scheme, amounts to the selection of: a) the model order by taking advantage of the orthogonalizing data property, of the lattice filter, and b) the lowpass detuning filter by progressively increasing its cut-off based on the accuracy of the identified model.

Table 1. ALSL–Algorithm

Initialization phase

$$\Delta_{m-1}(0) = 0 \; , \; F_{m-1}(0) = \delta$$
$$B_{m-1}(0) = \delta \; , \; \gamma_0(0) = 1$$
$$\kappa_{f,m}(0) = 0 \; , \; \kappa_{b,m}(0) = 0$$

At each instant n≥1 compute:

$$\eta_0(n) = \beta_0(n) = y(n)$$
$$F_0(n) = B_0(n) = \lambda F_0(n-1) + |y(n)|^2$$
$$\gamma_0(n-1) = 1$$

Recursive Computation Phase

For time $n = 1, 2, \ldots$ and order m=1,2,..., n_a

$$F_{m-1}(n) = \lambda F_{m-1,n-1} + \gamma_{m-1}(n-1)|\eta_{m-1}(n)|^2$$
$$B_{m-1}(n-1) = \lambda B_{m-1}(n-2) + \gamma_{m-1}(n-1)|\beta_{m-1}(n-1)|^2$$
$$\eta_m(n) = \eta_{m-1}(n) + \kappa^*_{f,m}(n-1)\beta_{m-1}(n-1)$$
$$\beta_m(n) = \beta_{m-1}(n-1) + \kappa^*_{b,m}(n-1)\eta_{m-1}(n)$$
$$\kappa_{f,m}(n) = \kappa_{f,m}(n-1) - \frac{\gamma_{m-1}(n-1)\beta_{m-1}(n-1)}{B_{m-1}(n-1)}\eta^*_m(n)$$
$$\kappa_{b,m}(n) = \kappa_{b,m}(n-1) - \frac{\gamma_{m-1}(n-1)\eta_{m-1}(n)}{F_{m-1}(n)}\beta^*_m(n)$$
$$\gamma_m(n-1) = \gamma_{m-1}(m-1) - \frac{\gamma^2_{m-1}(n-1)|\beta_{m-1}(n-1)|^2}{B_{m-1}(n-1)}$$

Table 2. Lattice to AR–Realization

$D_1(z) = \kappa_{f,1}$

Recursive Computation – For AR-order $m = 2, \ldots, n_a$.

Let $D_m(z) = d_0 + d_1 z + \ldots + d_m z^m$.

$$D^*_m(z) = d^*_{m-1} z^{m-1} + \ldots + d^*_0$$
$$D_m(z) = z\left[D_{m-1}(z) + \kappa^*_{f,m} D^*_m(z)\right] + \kappa_{f,m}$$

3.1 *Lattice Model Order Selection*

The primary advantage of using a lattice filter in the identification part is the trivial monitoring of the model's stability. The identified model is termed unstable, when the magnitude of the reflection coefficients exceeds unity, or when $|\kappa_{f,m}| > 1$. Other characteristics of a lattice filter include the reduced computational load for its implementation and its numerical robustness (Friedlander, 1982). The philosophy behind the "utilization" of additional cascade stages in the lattice filter structure is related to the orthogonalizing property of each stage. The mismatch between the system and the model response, measured by the forward and backward prediction errors ($f_i(n),\ b_i(n),\ i = 1, 2, \ldots$ parameters, shown in Figure 2) is smaller as the order increases. Every additional stage attempts to "whiten" the prediction error signals by orthogonalizing the filtered data; the perfect filter with order n_a has a prediction error equal to the corrupting noise ($f_{n_a}(t) = \nu(t)$). Based on this observation, several "rules" can be implemented to adjust the model's order.

3.2 *Lattice Filter Model Order Decrease*

If the filter's order exceeds that of the identified system, or $n_a \geq m$, then $\kappa_{f,i} = 0,\ i = m+1, \ldots$ Under the assumption of a current model order n_a, a learning rule of the following form can be used to reduce the model order

[Rule A] If $\max_{i\in[t-N_1,t]} |\kappa_{f,n_a}(i)| \leq \epsilon(\simeq 0^+)$, then $n_a \leftarrow n_a - 1$.

This rule amounts to ignoring all stages following the one with a small reflection coefficient over a sliding time window. The window span, N_1, is an ad–hoc selected parameter; large values indicate a willingness to await longer (slower convergence) prior to reducing the order, while at the same time ensuring that the reduction on the reflection coefficients is not to ill–conditioned data. For all practical purposes, it is not appropriate to let the model's order switch frequently. Furthermore, sufficient time must be provided for the reflection coefficients' convergence prior to testing the previous rule. Assume the time stamp t_{n_a}, as the event on which the model order changed to n_a. Then, the first rule is not allowed to be executed prior to elapsing N_2 samples after t_{n_a}, or

[Rule B] The proposition for [Rule A] cannot be tested prior to $t \geq t_{n_a} + N_2$.

In a similar manner, not only the magnitude of the reflection coefficient of the last stage must be small for reducing the model's order, but also its time-gradient (over a sliding window of length N_3 must be within small bounds for proper settling in its steady-state value. Therefore, the model order cannot be reduced unless the following rule is satisfied

[Rule C] If $\sum_{i=t-N_3}^{t} |\kappa_{f,n_a}(i) - \kappa_{f,n_a}(i-1)| \geq N_3 \times \text{constant}_C$ the model order cannot be reduced.

Overall, the aforementioned rules (A,B, and C) must be satisfied for reducing the model order from n_a to $n_a - 1$.

3.3 *Lattice Filter Model Order Increase*

Unlike classical estimation techniques (Haykin, 2001) which tend to overparametrize the model for capturing the dynamics of the unknown system, lattice filters can start with a small order (i.e., $n_a = 1$). After the lattice filter has been trained and its reflection coefficients have converged, the cumulative RMS–error (over a sliding window with length N_4) between its response $\hat{y}_{n_a}$ and the system response is computed. Depending on the magnitude of this quantity, the order can increase or remain at its current value. This is mirrored in the following rule

[Rule D] If $\sum_{i=t-N_4}^{t} [y(i) - \hat{y}_{n_a}(i)]^2 \geq N_4 \times \text{constant}_D$ the model order must be increased, or $n_a \leftarrow n_a + 1$.

Similar reasoning to stating Rule B necessitate the need to remain at a constant model order for a period of time N_5 prior to any possible increase, or

[Rule E] The proposition for [Rule D] cannot be tested prior to $t \geq t_{n_a} + N_5$.

Similarly, the time gradient signature of the last reflection coefficient must be tested prior to augmenting the model order, or

[Rule F] If $\sum_{i=t-N_6}^{t} |\kappa_{f,n_a}(i) - \kappa_{f,n_a}(i-1)| \geq N_6 \times \text{constant}_F$ the model order cannot be increased.

It is noteworthy, to state that the aforementioned six rules must be tested continuously (sample-by-sample basis) for proper adjustment of the system order. Therefore, at the expense of estimating the model order, the computational burden of the system identification routine is increased. Furthermore, several parameters $N_1, \ldots, N_6$ need to be selected on an ad-hoc basis; in a typical configuration $N_2 = N_5$ and $N_3 = N_6$.

3.4 *Detuning Filter Design*

Assume, that the identifier provides a model $\hat{G}_\mu$ of order μ for the system dynamics. The order μ can vary as stated in the previous paragraphs, thus affecting the H_2–controller's attributes, since $\tilde{q} = \hat{G}^{-1} = \frac{D_\mu(z)}{b_d z^{-d}}$. Furthermore, the 'detuning lowpass filter' $F(e^{\jmath\omega})$ is modified according to the μ–order of the identified system. The $F(z)$–filter design philosophy is to devise a scheme which progressively increases the bandwidth of the closed–loop system. Provided with an identified transfer function, the learning mechanism increases progressively the cutoff of filter F. Although this action increases the bandwidth of the closed loop system, it compromises the robust stability of the system. The robust stability of the system is satisfied if

$$|\hat{\eta} l_m| = |\hat{G}(e^{\jmath\omega})\tilde{q}(e^{\jmath\omega})F(e^{\jmath\omega})l_m| < 1$$

Therefore, the cutoff of this filter is progressively increased until the previous robust stability index has approached one. This progression results in a filter F^* with the largest bandwidth. If the bandwidth of the closed–loop system is satisfactory, then this learning process terminates. Let the identified transfer function computed for a certain number μ of reflection coefficients be denoted as $\hat{G}_\mu$. The proposed learning mechanism can be interpreted as:

[Rule G] For the identified system $\hat{G}_\mu$, design a sequence of controllers $q_\mu = \tilde{q}_\mu F_\mu$ that progressively increase the bandwidth of the closed–loop system.

In order to simplify the computational complexity of the algorithm, this filter is typically selected as a first order lowpass discrete filter

$$F_\mu(z) = \frac{\left(1 - e^{\lambda_\mu}\right) z}{z - e^{\lambda_\mu}} \ ,$$

where λ_μ corresponds to the cutoff frequency. The tuning algorithm starts with a small $\lambda_\mu \in [0, 0.5)$, which is progressively increased $\lambda_\mu = \lambda_\mu + \Delta\lambda_\mu$. The correction factor $\Delta\lambda_\mu (\geq 0)$ is adjusted based on the continuous monitoring of the squared tracking error over an M–sample sliding window $\sum_{t'=t-M}^{t} \left[r(t') - y(t')\right]^2$. If this error exceeds a certain number this can be an indication of marginal stability and the cutoff frequency is not increased.

4. SIMULATION STUDIES

The proposed adaptive IM-Controller is applied in simulation studies for controlling a system with a nominal transfer function $\tilde{G}(z)$ whose poles are located at $0.995\angle \pm 5°$ and $0.997\angle \pm 45°$. The output measurements are corrupted with a white noise signal $\nu(t)$ such that the SNR is 80 db.

4.1 *Open–Loop Lattice Filter Model Identification*

For purposes of investigating the efficiency of the lattice filter in the identification process, the open–loop system is excited with a white noise input sequence. The ad–hoc selected parameters are $N_j = 2, \ j = 1, 2, 4, 5$ and $N_i = 100, \ i = 3, 6$. The identified model order, n_a, as time progresses appears in Figure 3, while the convergence characteristics of the reflection coefficients is depicted in Figure 4.

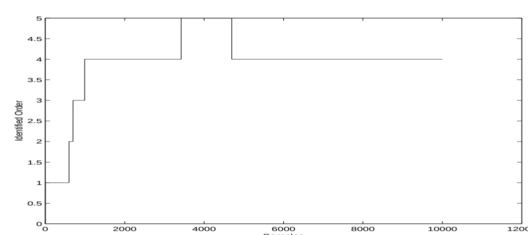

Fig. 3. Time history of identified model–order

Due to the inherent orthogonality of the lattice filters, when an order increase at stage n_a occurs, the previously defined reflection coefficients corresponding at the earlier stages $\kappa_{f,i}, \ i = 1, \ldots, n_a - 1$ remain constant. As expected, the first four lattice filter coefficients converge to their nominal values $\bar{\kappa}^{\text{nom}} = [-0.9890, 0.8150, -0.8878, 0.9841]^T$, while the fifth one settles to zero. It should be stated that during the instances where the model order changes the magnitude of the new reflection coefficient exceeds temporarily one, and typically converges within 5% of its nominal value in the subsequent 200 samples.

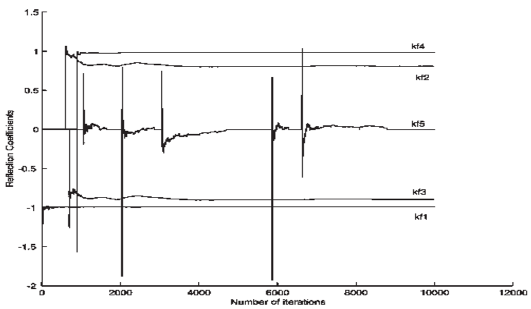

Fig. 4. ALSL-filter reflection coefficient convergence

4.2 *Closed–Loop AILC*

The overall suggested concept's efficiency is investigated in a closed–loop configuration, where the response of the system with the adaptive IM-Controller is shown in Figure 5. The variation of the model order appears in Figure 6. In comparison with the open–loop case, the model order raises significantly ($n_a^{\max} = 7$ for closed–loop versus $n_a^{\max} = 5$ for open–loop case) prior to settling down to its nominal value. This is attributed to the non utilization of an optimal prefilter and possible lack of persistently exciting signals.

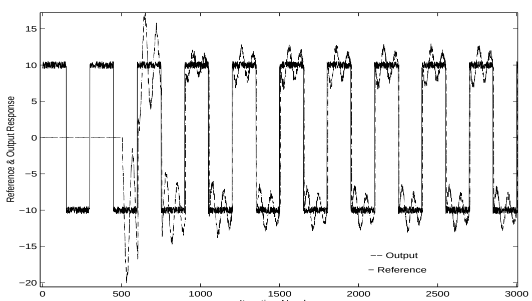

Fig. 5. System Response w./ AILController

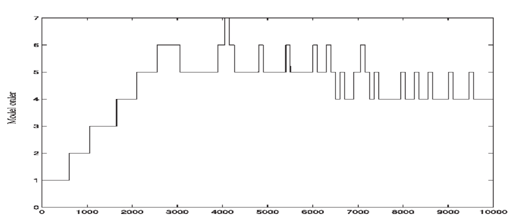

Fig. 6. Time history of model order change w./ AIMC

5. CONCLUSIONS

An adaptive IMC scheme was presented in this article. The unknown AR–system dynamics are identified through a lattice filter which estimates the process' reflection coefficients. The order of the system is inferred by monitoring the magnitude of the estimated coefficients and the cumulative error response. The IMController is the inverse of the estimated transfer function in cascade with a detuning filter. Simulation studies are used to illustrate the efficiency of the proposed scheme.

6. REFERENCES

Anderson, B.D.O. and R.L. Kosut (1991). Adaptive robust control: on-line learning. In: *Proceedings of the IEEE Conference on Decision and Control*. Vol. 1. Brighton, UK. pp. 297–298.

Datta, A. and L.Xing (1998). *Adaptive Internal Model Control*. Springer–Verlag. New York, NY.

Friedlander, B. (1982). Lattice Filters for Adaptive Processing. *Proceedings of the IEEE* **70**(8), 829–867.

Haykin, S. S. (2001). *Adaptive Filter Theory*. Prentice–Hall. Englewood Cliffs, NJ.

Lee, W.S., B.D.O. Anderson, R.L. Kosut and I.M.Y. Mareels (1993). A new approach to adaptive robust control. *International Journal of Adaptive Control and Signal Processing* **7**(2), 183–211.

M. Branicky (1998). Multiple Lyapunov Functions and Other Analysis Tools for Switched and Hybrid Systems. *IEEE Transactions on Automatic Control* **AC–43**, 475–482.

Morari, M. and E. Zafiriou (1989). *Robust Process Control*. Prentice-Hall. Englewood Cliffs, NJ.

Narenda, K.S. and J. Balakrishnan (1997). Adaptive control using multiple models. *IEEE Transactions on Automatic Control* **AC–42**, 171–187.

Rivera, D.E., J.F. Pollard and C.E. Garcia (1992). Control–relevant prefiltering: a systematic design approach and case study. *IEEE Transactions on Automatic Control* **37**(7), 964–974.

Silva, G. and A. Datta (2001). Adaptive internal model control: The discrete case. *International Journal of Adaptive Control and Signal Processing* **15**(1), 15–36.

Sun, Z. and D. Zheng (2001). On reachability and stabilization of switched linear systems. *IEEE Transactions on Automatic Control* **AC–46**, 291–295.

Y. Zhu and P. van der Bosch (2001). Optimal closed-loop identification test design for internal model control. *Automatica* **36**(8), 1237–1241.