'93 한국자동제어학술회의논문집 (1993.10.20~22)

ARMA 모델을 이용한 적응 모델예측제어에 관한 연구

°이종구, 김석준, 박선원
한국과학기술원 화학공학과

Adaptive Model Predictive Control Using ARMA Models

°Jongku Lee, Sukjoon Kim, and Sunwon Park
Department of Chemical Engineering, KAIST

Abstract

An adaptive model predictive control (AMPC) strategy using auto-regression moving-average (ARMA) models is presented. The characteristic features of this methodology are the small computer memory requirement, high computational speed, robustness, and easy handling of nonlinear and time varying MIMO systems. Since the process dynamic behaviors are expressed by ARMA models, the model parameter adaptation is simple and fast to converge. The recursive least square (RLS) method with exponential forgetting is used to trace the process model parameters assuming the process is slowly time varying. The control performance of the AMPC is verified by both comparative simulation and experimental studies on distillation column control.

1. Introduction

During the last 15 years, Model Predictive Control (MPC) has been recognized as a prominent practical control methodology and demonstrated to perform well in a wide range of process control applications. There are some representative MPC such as Model Algorithmic Control (MAC) [Richalet, 1978; Mehra et al., 1979], Dynamic Matrix Control (DMC) [Culter and Ramaker, 1979], and Internal Model Control (IMC) [Garcia and Morari, 1982] that have been well received by industry. Most MPC techniques use the weighting sequence models such as the step or impulse response models that are physically intuitive, and they can handle constraints explicitly when they are combined with on-line optimization code.

The weighting sequence models require little knowledge of process characteristics (i.e. system order or time delay) and can be made by simply giving a step or an impulse process input to the process. However, they require too many parameters to describe the processes, which cause heavy computational burden and large computer memory requirement [Morari and Lee, 1991]. Moreover, their uses are restricted to the open-loop stable processes, and it is not easy to update the process models due to large amount of process model parameters.

Several model predictive controllers based on the parametric input-output models have been developed from the concept of adaptive controllers [Lee and Lee, 1983; Ydistie, 1984; Cauwenberghe and De Keyser, 1985; Clarke et al., 1987]. In contrast to the weighting sequence models, the parametric input-output models have some positive aspect because they are modeled by use of small process model parameters and can estimate the model parameters with ease even when the dynamic behaviors of processes are changed. The Extended Horizon Adaptive Control (EHAC) [Ydstie, 1984] and Extended Prediction Self Adaptive Control (EPSAC) [Cauwenberghe and De Keyser, 1985] use the Auto-Regressive Moving-Average with eXogenous input (ARMAX) model. But the ARMAX model is not satisfactory for dealing with the offset problem, especially when the load disturbance is rapidly varying [Clarke, 1983]. Clarke et al. (1987a) developed the Generalized Predictive Controller (GPC) based on the Controlled Auto-Regressive Integrated Moving-Average (CARIMA) which are known to be useful to eliminate the offset. Also, this can be applied to a nonminimum-phase or an open-loop unstable plant which is practically difficult system to control by the existing model predictive controllers. Shah et al. (1987) and Kinnaert (1989) extended the GPC to MIMO systems.

In the GPC algorithm, the future output prediction and the predictive control law are developed by solving the Diophantine equation of the CARIMA plant model, which increases the computational burden, especially, in the case of MIMO processes. Moreover, if the processes are time-varying or show severe nonlinear behavior, the model parameter adaptation is required to trace the process dynamic behavior and the numerous Diophantine equation should be solved whenever the process model is updated, which makes GPC less robust numerically [Morari and Lee, 1991]. From the above reason, the applications of GPC to the practical chemical processes are few.

In this paper, we propose new Adaptive Model Predictive Control (AMPC) which is based on ARMA models. Its control law is almost the same with DMC or GPC, but the output prediction part is simplified by using ARMA models. Its prediction law is based on the step response model that is calculated from the ARMA models. Therefore, it does not require solving the Diophantine equation which is the main difference from the GPC. The merits of AMPC are easy application to complex MIMO processes, small computer memory requirement, high computational speed with numerical robustness, and applicability to nonlinear and time varying systems.

2. Adaptive Model Predictive Control Algorithm

2.1 Models and Parameter Estimation

Most SISO processes can be accurately approximated in the local operating region by the linear discrete-time model as follows :

$$A(q^{-1})y(k) = q^{-k_d}B(q^{-1})u(k) + d(k) + \upsilon(k) \quad (1)$$

where

$$A(q^{-1}) \equiv 1 + \sum_{i=1}^{n_a} a_i q^{-i} \quad (2)$$

$$B(q^{-1}) \equiv \sum_{i=1}^{n_b} b_i q^{-i} \quad (3)$$

q^{-1} is the unit delay operator, k is the present time, k_d is the system time delay, $d(\cdot)$ is the load disturbance, $\upsilon(\cdot)$ is the random noise, $y(\cdot)$ is the measured process output, $u(\cdot)$ is the control input. This model form (ARMA) is very useful for model parameter identification. In the literature, $\upsilon(k)$ has been considered to be of moving average form :

$$\upsilon(k) \equiv \begin{cases} C(q^{-1})\xi(k) & \text{(ARMAX model)} \\ C(q^{-1})\xi(k)/\Delta & \text{(CARIMA model)} \end{cases} \quad (4)$$

where

$$C(q^{-1}) \equiv \sum_{i=0}^{n_c} c_i q^{-i} \quad (5)$$

$\xi(\cdot) \equiv$ uncorrelated random sequence

$\Delta \equiv$ differencing operator $(1-q^{-1})$

The information about $\upsilon(k)$ is very important to overcome the model parameter convergence problem which occasionally occurs in model parameter identification. However, in practical situations, there is no prior knowledge about the weighting of the random sequence, and even though the polynomial $C(q^{-1})$ is known, the term $\upsilon(k)$ is hardly used to design the model predictive controllers. In the design of model predictive controllers, the most important information is the relationship between the control inputs and process outputs.

Most practical processes are nonlinear, while the process models are approximated as linear models. Thus it is necessary to update the model parameters as the status of the process changes even when the process is not time-varying. The recursive least square (RLS) algorithm is one of the most widely used recursive identification methods. It is robust and easily implemented. A disadvantage with the least square estimate is that in general $\upsilon(k)$ and process measurements will be found to be correlated, and then the model parameters will not converge to its true values. However, practically, the convergence of the model parameters is rather important than the accuracy of them since the desirable feature of the MPC is robustness to modeling errors. Therefore, if the signal/noise ratio is not too small, the recursive least square method with exponential forgetting can be used for the adaptation of the time-varying model parameters.

2.2 Controller Design

The control law of AMPC is nearly the same as that of DMC, but the output prediction is obtained by ARMA models. Consider a discrete "step response model" which is used in the basic Dynamic Matrix Control derivation as follows [Garcia and Morshedi, 1986] :

$$y(k+i) = y*(k+i) + \sum_{j=1}^{i} s_j \Delta u(k+i-j) + d(k)$$

$$\text{for } i = 1, 2, \cdots, P \tag{6}$$

where

$P \equiv$ prediction horizon

$$y*(k+i) = \sum_{j=i+1}^{M} s_j \Delta u(k+i-j), \text{ where}$$

M is the model horizon (7)

$s_j \equiv$ step response coefficients

$\Delta u(\cdot) \equiv$ change in input

In DMC algorithm, memory requirement is large in order to store the history of inputs moves and the step response model since $y*(\cdot)$ is calculated by the step response model. That causes the insufficient computer memory problem and heavy computation burden when DMC is applied to complex MIMO processes. However, using the ARMA model, we can calculate $y*(\cdot)$ as follows :

$$y*(k+i) = -\sum_{j=1}^{n_a} \hat{a}_j y*(k+i-j) + \sum_{j=1}^{n_b} \hat{b}_j u*(k+i-j-k_d)$$

$$i = 1, 2, \ldots, P \tag{8}$$

where

$$y*(\cdot) \equiv y(\cdot) \quad \text{if } i-j \le 0$$

$$u*(\cdot) \equiv \begin{cases} u(\cdot) & \text{if } i-j-k_d < 0 \\ u(k-1) & \text{if } i-j-k_d \ge 0 \end{cases}$$

Since n_a, n_b « M, the computational burden for the predicted process output calculation is far less than that of DMC.

Next, if the model parameters are updated by the parameter estimator, we have to update the step response coefficients, $s_1, s_2, \cdots, s_P$ as follows :

$$s_i = -\sum_{j=1}^{n_a} \hat{a}_j s_{i-j} + \sum_{j=1}^{n_b} \hat{b}_j u(i-j-k_d), \quad i = 1, 2, \ldots, P \tag{9}$$

where

$$s_{i-j} = 0 \quad \text{if } i-j \le 0$$

$$u(\cdot) \equiv \begin{cases} 0 & \text{if } i-j-k_d < 0 \\ 1 & \text{if } i-j-k_d \ge 0 \end{cases}$$

and the dynamic matrix is made as follows :

$$\begin{bmatrix} s_1 & 0 & \cdots & 0 \\ s_2 & s_1 & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & s_1 \\ \cdot & \cdot & \cdots & \cdot \\ s_P & s_{P-1} & \cdots & s_{P-N+1} \end{bmatrix} \tag{10}$$

where $N \equiv$ control horizon

The process output prediction is done by only using the ARMA model, which is numerically stable. Then, the predictive control law is given by minimizing the following objective function.

$$\min \Phi(\Delta u) = \sum_{i=1}^{P} \gamma^2(i) \left[(y(k+i) - y_{sp}(k+i) \right]^2$$

$$+ \sum_{j=1}^{N} \lambda^2(j) \left[\Delta u(k+j-1) \right]^2 \tag{11}$$

$$\text{s.t. } u_{\min} \le u(k-1) + \sum_{l=1}^{i} \Delta u(k+l-1) \le u_{\max} \tag{12}$$

$$y_{\min} \le y(k) + \sum_{l=1}^{i} s_l \Delta u(k+l-1) \le y_{\max} \tag{13}$$

where $y_{sp}(k+i)$ is the future setpoints, γ and λ are the weights on the process output error and on the change in the control input, respectively. In this step, future optimal inputs $u(k+i-1)$ for i = 1, 2, ... ,N are calculated by a QP algorithm and only the first input $u(k)$ is implemented.

2.3 Extension to MIMO systems

In linear systems, most MIMO processes are expressed in the transfer function matrix form as follows :

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) & \cdots & G_{1m}(s) \\ G_{21}(s) & G_{22}(s) & \cdots & G_{2m}(s) \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ G_{n1}(s) & G_{n2}(s) & \cdots & G_{nm}(s) \end{bmatrix}$$

$$= \begin{bmatrix} \frac{N_{11}(s)}{D_{11}(s)} & \frac{N_{12}(s)}{D_{12}(s)} & \cdots & \frac{N_{1m}(s)}{D_{1m}(s)} \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \frac{N_{n1}(s)}{D_{n1}(s)} & \frac{N_{n2}(s)}{D_{n2}(s)} & \cdots & \frac{N_{nm}(s)}{D_{nm}(s)} \end{bmatrix} \tag{14}$$

where $G_{ij}(s)$ is the transfer function relating the i^{th} output to the j^{th} input, n and m are the number of the process outputs and control inputs, respectively. In order to design the AMPC for the above MIMO system, we modify the transfer functions a little to make simple ARMA models. In each row, we find the lowest order common denominator of the transfer functions and based on it, rearrange the numerators to fit the original transfer functions.

$$G(s) = \begin{bmatrix} \frac{\overline{N_{11}}(s)}{D_1(s)} & \frac{\overline{N_{12}}(s)}{D_1(s)} & \cdots & \frac{\overline{N_{1m}}(s)}{D_1(s)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\overline{N_{n1}}(s)}{D_n(s)} & \frac{\overline{N_{n2}}(s)}{D_n(s)} & \cdots & \frac{\overline{N_{nm}}(s)}{D_n(s)} \end{bmatrix} \tag{15}$$

where $D_i(s)$ is the lowest order common denominator among the transfer functions in the i^{th} row and $\overline{N_{ij}}(s)$ is the rearranged numerator to satisfy the ij^{th} original transfer function. Since the denominators on the same row are common, process outputs are expressed as follows :

$$\begin{aligned} Y_1(s) &= \frac{\overline{N_{11}}(s)\,U_1(s) + \cdots + \overline{N_{1m}}(s)\,U_m(s)}{D_1(s)} \\ &\vdots \\ Y_n(s) &= \frac{\overline{N_{n1}}(s)\,U_1(s) + \cdots + \overline{N_{nm}}(s)\,U_m(s)}{D_n(s)} \end{aligned} \tag{16}$$

where $Y_i(s)$ is the i^{th} output, and $U_j(s)$ is the j^{th} input in the Laplace domain. Then the above transfer function models can be converted to make the n-ARMA models.

$$\begin{bmatrix} A_1(q^{-1}) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & A_n(q^{-1}) \end{bmatrix} Y(k) = \begin{bmatrix} q^{-k_d^{11}} B_{11}(q^{-1}) & \cdots & q^{-k_d^{1m}} B_{1m}(q^{-1}) \\ \vdots & \ddots & \vdots \\ q^{-k_d^{n1}} B_{n1}(q^{-1}) & \cdots & q^{-k_d^{nm}} B_{nm}(q^{-1}) \end{bmatrix} U(k) \tag{17}$$

where

$$A_i(q^{-1}) = 1 + a_1^i q^{-1} + \dots + a_{n_{A_i}}^i q^{-n_{A_i}}$$

$$B_{ij}(q^{-1}) = b_1^{ij} + b_2^{ij} q^{-1} + \dots + b_{n_{B_{ij}}}^{ij} q^{-n_{B_{ij}}}$$

$$Y(k) \equiv [\, y_1(k), y_2(k), \dots, y_n(k) \,]^T$$

$$U(k) \equiv [\, u_1(k), u_2(k), \dots, u_m(k) \,]^T$$

and k_d^{ij} is the time delay of the ij^{th} transfer function.

The controller design procedures for MIMO systems are straightforward. The output predictions and the formulation of Dynamic Matrix are made through the repetitive vector multiplication for each process output as in the case of the SISO system. Also, the predictive control law is exactly the same as in the SISO case.

3. Experimental Study with a pilot distillation column

The top and bottom temperatures of a pilot scale MeOH distillation column are controlled by manipulating the heat to the reboiler and the reflux flow rate. This is a standard dual temperature control problem that has been attractive for control study due to strong interactions and nonlinearity at high purity range. These characteristics motivate the use of adaptive model predictive controllers such as AMPC.

System dynamics

To characterize the dynamics of the distillation column, we performed several step tests within the operating range. Figure 1 shows the responses of the top and bottom temperatures (y_1 and y_2) to a step up and a step down in the reflux flow rate (u_1) and the reboiler heat input (u_2). The figure shows the asymmetry of the output responses to the same magnitude of input changes in the opposite directions, which is the typical nonlinear behavior of distillation columns. We approximated open-loop output responses as first order linear models with time delay. The approximated process models are as follows :

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{-1.85e^{-0.33s}}{12.0s+1} & \frac{1.2e^{-0.67s}}{6.0s+1} \\ \frac{-0.9e^{-s}}{28s+1} & \frac{3.56^{-0.33s}}{15.0s+1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{18}$$

Since the transfer function model is approximated one, the inherent model/plant mismatch exists.

Designs of AMPC and QDMC

The AMPC and QDMC were applied to the distillation column control. Parameters to be specified in the design of the QDMC are : sampling time, prediction horizon or optimization horizon, control horizon, move suppression factor, and boundary constraints. In the case of the AMPC, the process model order, forgetting factor, and dead zone should be specified additionally. The reflux flow rate (u_1) is bounded between 0 and 150 ml/min and the reboiler heat (u_2) between 60 and 90 % of maximum heat input. Table 1 lists the design parameters of the AMPC and QDMC. The tuning parameters were decided through numerous experimental results of the QDMC. Among them, we chose the tuning parameters based on the best control performance under the integral of the squared error (ISE) criterion. The AMPC has the same tuning parameters as the QDMC except for the prediction horizon and the design parameters about the process model parameter identification.

Table 1. The tuning parameters of the AMPC and QDMC

AMPC		QDMC	
sampling time	20 sec	sampling time	20 sec
prediction horizon	30	model horizon	180
control horizon	5	prediction horizon	48
move suppression	0.1	control horizon	5
process model order (A,B)	(3,3)	move suppression	0.3
forgetting factor	0.9		
dead zone	±0.1 ℃		

Experimental Results

The servo responses of the AMPC and QDMC are compared. We changed the setpoints of the top and bottom temperatures so that the process outputs could cover the wide operating range. Figure 2 shows the servo responses of the QDMC and AMPC. In the high purity region ($50<t<70$), the top temperature follows its setpoint very slowly and the bottom temperature deviate severely from its setpoint in the case of the QDMC, but the AMPC shows much better control performance compared with the results of the QDMC. Next, to test the unknown disturbance-rejection performance, we change the feed flow rate from 600 cc/min to 500 cc/min at t=3 min and restore it at t=23 min. The AMPC shows faster rejection performance than the QDMC (Figure 3). Table 2 lists the ISE of the AMPC and QDMC. The ISEs of the bottom temperature in the case of AMPC are far smaller than those in the case of QDMC. From the above results, we can see that the AMPC is excellent in the setpoint-tracking and disturbance-rejection when it is applied to nonlinear processes. The calculation time of the AMPC is faster than that of QDMC even though the AMPC includes the parameter adaptation algorithm, and the size of execution file of AMPC is much smaller than that of QDMC.

Table 2. ISE of the QDMC and AMPC

		Top temp.	Bottom temp.	Total sum
Setpoint Tracking	QDMC	192.79	191.59	384.38
	AMPC	196.88	107.16	304.04
Disturb. Rejection	QDMC	8.78	114.10	122.88
	AMPC	7.49	39.59	47.08

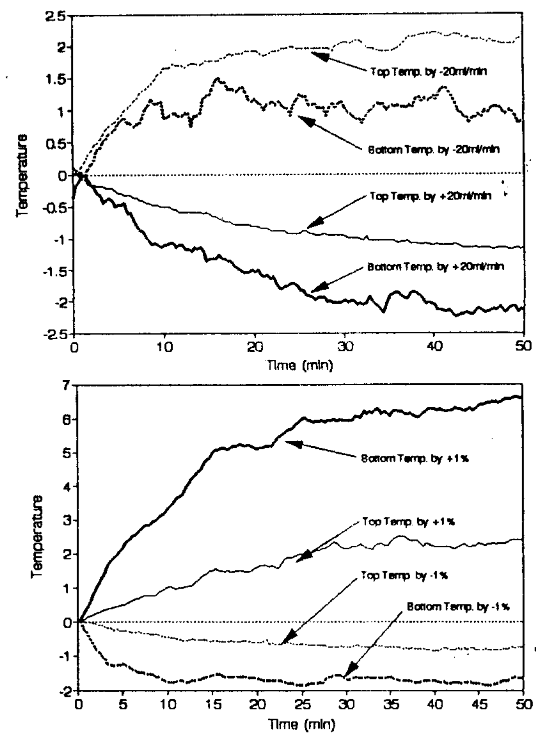
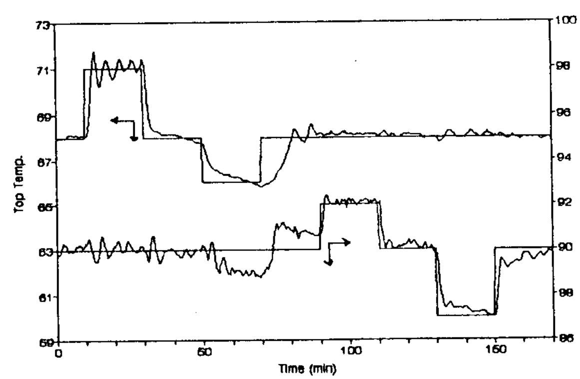
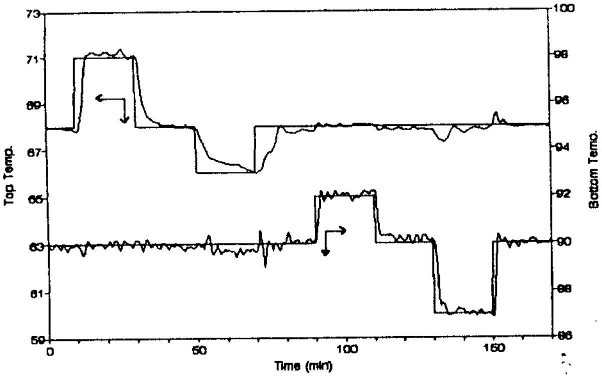


Figure 1. Open-loop step responses

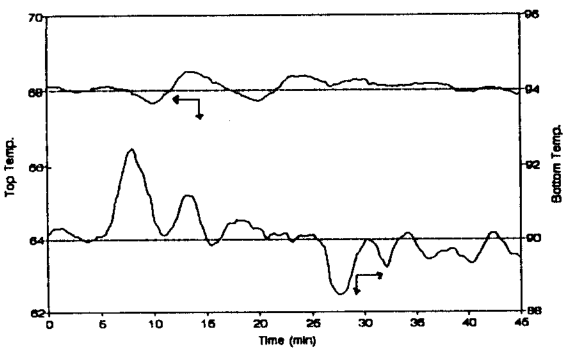


(a)

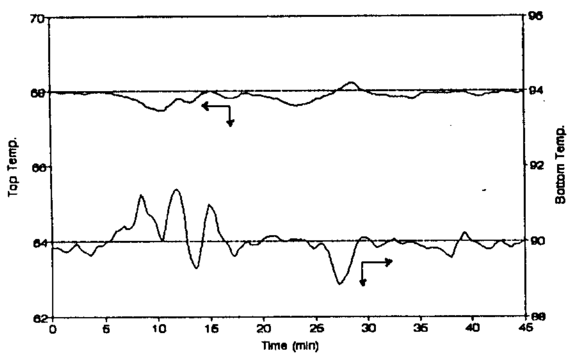


(b)

Figure 2. Setpoint-tracking performance: (a) QDMC (b) AMPC



(a)



(b)

Figure 3. Disturbance-rejection performance: (a) QDMC (b) AMPC

4. Conclusions

An Adaptive Model Predictive Controller (AMPC) using ARMA models has been presented. The characteristic features of the proposed methodology are small computational memory requirement, high computational speed, robustness, and easy handling of nonlinear and time-varying MIMO processes while showing good control performance. Since the AMPC uses ARMA models, the calculation of process output prediction is simple and does not require solving the Diophantine equation which is required in the GPC algorithm. Therefore, the AMPC algorithm is numerically robust and its computational speed is very fast even though it includes the parameter adaptation algorithm. The control performance of the AMPC has been verified by the comparative simulation studies and experimental application to a pilot scale distillation column. We expect the AMPC to show better control performance than other MPC techniques for nonlinear processes control.

References

Aström, K. J. and B. Wittenmark, "Adaptive Control," Addison-Wesley (1989).

Byun, D. G., and W. H. Kwon, "Predictive Control: A review and Some New Stability Results," T. J. McAvoy, Y. Arkun, and E. Zafiriou, eds., Proc. IFAC Workshop on Model Based Process Control, Pergamon Press, Oxford, 81 (1988).

Clarke, D. W., A. J. F. Hodgson, and P. S. Tuffs, "Offset Problem and k-Incremental Predictors in Self-Tuning Control," IEE Proc., 130, Pt.D, no.5, Sept. (1987).

Clarke, D. W., C. Mohtadi, and P. S. Tuffs, "Generalized Predictive Control - Part I. The Basic Algorithm," Automatica, 23, 137 (1987).

Cutler, C. R. and B. L. Ramaker, "Dynamic Matrix Control-A Computer Control Algorithm," JACC, San Francisco (1979).

Garcia, C. E. and A. M. Morshedi, "Quadratic Programming Solution of Dynamic Matrix Control(QDMC)," Chem. Eng. Commun. 46, 73 (1986).

Garcia, C. E., D. M. Prett, and M. Morari, "Model Predictive Control: Theory and Practice-a Survey," Automatica, 25, 335 (1989).

Garcia, C. E. and M. Morari, "Internal model control-1. A unifying review and some new results," Ind. Engng. Chem. Process Des. Dev., 21, 308 (1982).

Hernández, E. and Y. Arkun, "Control of Nonlinear Systems Using Polynomial ARMA Models," AIChE J. 39, 446 (1993).

Kinnaert, M., "Adaptive generalized predictive controller for MIMO systems," Int. J. Control, 50, 161 (1989).

Lee, K. S. and W. K. Lee, "Extended discrete-time multivariable adaptive control using long-term predictor," Int. J. Control, 38, 49 (1983).

Limqueco, L. C., and J. C. Kantor, "Nonlinear Output Feedback Control of an Exothermic Reactor," Comp. Chem. Eng., 14, 427 (1990).

Ljung, L, "System Identification : Theory for the User," Prentice-Hall, Englewood Cliffs, NJ (1987).

Mehra, R. K., R. Rouhani, A. Rault, and J. G. Reid, "Model Algorithmic Control : Theoretical Results on Robustness," Proc. JACC, 387 (1979).

Morari, M. and J. H. Lee, "Model Predictive Control : the Good, the Bad and the Ugly," CPC IV, 419 (1991).

Ydistie, B. E., "Extended horizon adaptive control," 9th Triennial IFAC World Congress, Budapest, Hungary (1984).