DTU Library

Robust Comparison of the Linear Model Structures in Self-tuning Adaptive Control

Zhou, Jianjun; Conrad, Finn

Publication date:
1989

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Robust Comparison of the Linear Model Structures in Self-tuning Adaptive Control

Zhou Jianjun   Finn Conrad

Control Engineering Institute
Building 424
Technical University of Denmark
DK-2800 Lyngby Denmark

Abstract

The Generalized Predictive Controller (GPC) is extended to the systems with a generalized linear model structure which contains a number of choices of linear model structures. The Recursive Prediction Error Method (RPEM) is used to estimate the unknown parameters of the linear model structures to constitute a GPC self-tuner. Different linear model structures commonly used are compared and evaluated by applying them to the extended GPC self-tuner as well as to the special cases of the GPC, the GMV and MV self-tuners. The simulation results show how the choice of model structure affects the input- output behaviour of self-tuning controllers.

1 Introduction

It is well known that a true dynamic system can not be represented exactly within the chosen model sets, and the real system to be identified is usually more complicated than the chosen model. In the Self-Tuning Adaptive Control (STAC) literature, most controller designers use a Controlled Autoregressive - Moving Average (CARMA) model to describe the processes with stochastic disturbance. One might ask if the CARMA model is the best choice and how the choice of model structure affects the performance and overall reliability of adaptive control. This paper tries to answer these questions and deals with the comparison of robustness to various linear model structures.

2 Various Linear Model Structures

We assume that the locally-linearized plant considered can be modelled by a generalized linear model structure

$$A(q^{-1})y(t) = \frac{B(q^{-1})}{F(q^{-1})}u(t-k_{min}) + \frac{C(q^{-1})}{D(q^{-1})}e(t) + d$$

$$(2.1)$$

where y(t) and u(t-k_{min}) are the output and the input sequences respectively, k_{min} is the minimum expected delay-time which can

be set to unity if it is not available. e(t) is a non-measurable, normally distributed white noise. $A(q^{-1})$, $B(q^{-1})$, $C(q^{-1})$, $D(q^{-1})$ and $F(q^{-1})$ are polynomials in the backward shift operator q^{-1} with degree n_a, n_b, n_c, n_d, and n_f, respectively. It is assumed that $A(q^{-1})$, $C(q^{-1})$, $D(q^{-1})$ and $F(q^{-1})$ are monic. The scalar d represents an offset due to either the local linearization of an essentially non-linear process or the presence of non-zero mean load disturbances. Tuffs and Clark (1984) have argued that the integrated CARMA model, i.e. CARIMA model, is most suitable for dealing with non-stationary disturbances. We extend their idea to the generalized linear model (2.1). Thus an integrated generalized linear model can be obtained:

$$A(q^{-1})\Delta y(t) = \frac{B(q^{-1})}{F(q^{-1})}\Delta u(t-k_{min}) + \frac{C(q^{-1})}{D(q^{-1})}e(t)$$

$$(2.2)$$

where $\Delta = (1-q^{-1})$

The linear model (2.2) is too general for most practical purposes. One or several of the five polynomial would be fixed to unity in application. The model structures used commonly in control design are shown in the following:

$$A(q^{-1})\Delta y(t) = B(q^{-1})\Delta u(t-k_{min}) + e(t) \qquad (2.3)$$

$$A(q^{-1})\Delta y(t) = B(q^{-1})\Delta u(t-k_{min}) + C(q^{-1})e(t)$$

$$(2.4)$$

$$A(q^{-1})\Delta y(t) = B(q^{-1})\Delta u(t-k_{min}) + \frac{1}{D(q^{-1})}e(t)$$

$$(2.5)$$

$$A(q^{-1})\Delta y(t) = B(q^{-1})\Delta u(t-k_{min}) + \frac{C(q^{-1})}{D(q^{-1})}e(t)$$

$$(2.6)$$

$$\Delta y(t) = \frac{B(q^{-1})}{F(q^{-1})}\Delta u(t-k_{min}) + e(t) \qquad (2.7)$$

$$\Delta y(t) = \frac{B(q^{-1})}{F(q^{-1})}\Delta u(t - k_{\min}) + \frac{1}{D(q^{-1})}e(t) \qquad (2.8)$$

$$\Delta y(t) = \frac{B(q^{-1})}{F(q^{-1})}\Delta u(t - k_{\min}) + C(q^{-1})e(t) \qquad (2.9)$$

$$\Delta y(t) = \frac{B(q^{-1})}{F(q^{-1})}\Delta \ddot{u}(t - k_{\min}) + \frac{C(q^{-1})}{D(q^{-1})}e(t) \qquad (2.10)$$

For the sake of simplicity in the following sections, we use notation: LM(A,B,C,D,F) to denote the linear model (2.2). Similar notations are used for other model sets, e.g, denoting the linear model (2.4) as LM(A,B,C).

3 Generalized Predictive Control With A Generalized Linear Model Structure

Control Design

The cost function to be considered is of the form

$$J(N_1,N_2) = E\left\{ \sum_{i=N_1}^{N_2} \phi_i^2(t) + \mu \sum_{i=1}^{N_2}(Q_i(q^{-1})u(t+i-k_{\min})^2 \right\}$$

$$(3.1)$$

where $\phi_i(t) = P(q^{-1})y(t+i) - R(q^{-1})w(t+i - K_{\min})$

where E(.) is the expectation operator conditioned on data up to time t, $P(q^{-1})$, $R(q^{-1})$ and $Q_i(q^{-1})$ (i=1,2,...,N_2) are polynomials or rational transfer functions. In addition, a non-negative scalar μ is the weighting factor of the control signal and w(t) is the set points. It is assumed that $N_1 = K_{\min}$, $N_2 = N$, $Q_i = 1 - q^{-1}$, (i= 1, 2, ..., N), P and R are finite polynomials with $p_0 = 1$, $r_0 = 1$.

Combining the process model (2.3) with the Diophantine identity

$$CP = D\Delta AL_i + q^{-1}G_i \qquad (3.2)$$

where L_i and G_i are polynomials with $\deg(L_i) = i-1$, and $\deg(G_i) = \max(n_c + n_p - i, n_d + n_a)$, respectively, we obtain

$$\phi_i(t) = C^{-1}F^{-1}DL_iB\Delta u(t+i-k_{\min}) + L_ie(t+i)$$

$$+ C^{-1}G_iy(t) - Rw(t+i-k_{\min}) \quad i=k_{\min},...,N$$

The first term on the right-hand side can be partitioned into past (known) and future (unknown) parts, by using a second polynomial partition:

$$DL_iB = CFE_j + q_j^{-i} \qquad (3.3)$$

with $j = i - k_{\min} + 1$, $Deg(E_j) = j-1$

$$Deg(\Gamma_j) = \max(n_d + n_b + k_{\min} - 2, n_c + n_f - 1)$$

Then $\phi_i(t) = E_j\Delta u(t+i-k_{\min}) + \Gamma_j C^{-1}F^{-1}\Delta u(t-1) + L_ie(t+i)$

$$+ C^{-1}G_iy(t) - Rw(t+i-k_{\min})$$

A set of $\phi_i(t)$ for various i given by i=$k_{\min}$, ..., N, can be stacked together and written in vector form as

$$\Phi^T = [\phi_{kmin}(t),...,\phi_N(t)] \qquad (3.4)$$

i.e. $\Phi = EU + C^{-1}F^{-1}\Delta u(t-1) + C^{-1}Gy(t) + L - RW$

$$(3.5a)$$

where

$$E = \begin{bmatrix} e_0 & 0 & \cdots & 0 \\ e_1 & e_0 & \cdots & 0 \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ e_{N-kmin} & e_{N-kmin+1} & \cdots & e_0 \end{bmatrix} \qquad (3.5b)$$

$$U^T = [\Delta u(t), \Delta u(t+1),...,\Delta u(t+N-k_{\min})] \qquad (3.5c)$$

$$\Gamma^T = [\Gamma_1, \Gamma_2,...,\Gamma_{a-k_{min}+1}] \qquad (3.5d)$$

$$G^T = [G_{kmin}, G_{kmin+1},...,G_N] \qquad (3.5e)$$

$$L^T = [L_{kmin}e(t+k_{\min}), L_{kmin+1}e(t+k_{\min}+1),...,L_Ne(t+N)]$$

$$(3.5f)$$

$$W^T = [w(t), w(t+1),...,w(t+N-k_{\min})] \qquad (3.5g)$$

where the parameters e_i (i=0,1, ..., N-$k_{\min}$) are coefficients of the polynomial $E_{N-k_{min}+1}$.

From (3.4) and (3.5), the cost function can be simplified as

$$J(k_{\min},N) = E\{\Phi^T\Phi + \mu U^TU\} \qquad (3.6)$$

The minimization of the cost function (3.6) with respect to U results in the identity:

$$(E^TE+rI)U = E^T[RW - C^{-1}F^{-1}\Gamma\Delta u(t-1) - C^{-1}Gy(t)$$

$$(3.7)$$

Defining a $(N-k_{\min}+1) \times (N-k_{\min}+1)$ dimension matrix, X, as the solution of the linear equation set:

$$(E^TE+\mu I)X = E^T \qquad (3.8)$$

and the first row of X as

$$K^T = [k_1 k_2 ... k_{N-kmin+1}]$$

then $\Delta u(t) = K^T[RW - C^{-1}F^{-1}\Gamma\Delta u(t-1) - C^{-1}Gy(t)]$

$$(3.9)$$

The above equation can be simplified further:

$$\Delta u(t) = \frac{K'RCFw(t+N-k_{\min}) - K^TFGy(t)}{CF + K^T\Gamma q^{-1}} \qquad (3.10)$$

where

$$K' = k_{N-kmin+1} + k_{N-kmin}q^{-1} + ... + k_1q^{-N+kmin} \qquad (3.11)$$

2201

To reduce computation expense and avoid singularity of the coefficient matrix $E^T E + \mu I$ in (3.8), a control horizon, N_U is recommended to use [see [2] for detail].

Closed-Loop Equation

The closed-loop equation can be obtained from (2.1) and (3.10)

$$y(t) = \frac{BK'RCq^{-k_{min}}w(t+N-k_{min}) + (CF + K^T\Gamma q^{-1})C/De(t)}{\Delta A(CF + K^T\Gamma q^{-1}) + BK^T Gq^{-k_{min}}}$$

(3.12)

The closed loop characteristic equation is given by

$$\Delta A(CF + K^T\Gamma q^{-1}) + BK^T Gq^{-k_{min}} = C(\Delta AF + q^{-1}K^T H)$$

(3.13)

where $H^T = [H_1 H_2 ... H_{N-k_{min}}]$

H_i $(i=1,...,N-k_{min})$ are calculated by solving the equations:

$$BP = \Delta AFE_j + q^{-j}H_j \qquad j = 1,...,N - k^{min}$$

The Generalized Minimum Variance (GPC) controller for a plant with known delay-time k can be seen to be a special case of the GPC in which k_{min}, N_1 and N_2 are set to k and only one control signal is weighted. Letting weighting scalar μ be zero, we obtain the Minimum Variance (MV) controller.

4 Overview of the RPEM and RPLR

The design of STAC controller requires that the linear model parameters be know at all times. The RPEM is used to estimate the unknown parameters of various linear model structures.

Introducing variables:

$$w(t) = \frac{B(q^{-1})}{F(q^{-1})}\bar{u}(t) \quad (4.1a) \qquad \tau(t) = A(q^{-1})y(t) - w(t) \quad (4.1b)$$

$$\epsilon(t) = \frac{D(q^{-1})}{C(q^{-1})}\tau(t) \quad (4.1c)$$

the linear model (2.2) can be written as

$$\bar{y}(t) = [1 - A(q^{-1})]\bar{y}(t) + B(q^{-1})\bar{u}(t) + [1 - F(q^{-1})]w(t)$$

$$+ [C(q^{-1}) - 1]\epsilon(t) + [1 - D(q^{-1})]\gamma(t) + e(t)$$

$$= \Phi^T(t)\theta + e(t)$$

(4.2)

where $\quad \bar{y}(t) = y(t) - y(t-1)$

$$\bar{u}(t) = u(t) - u(t-1)$$

$$\theta^T = [a_1...a_{n_a} b_1...b_{n_b} f_1...f_{n_f}$$

$$c_1...c_{n_c} d_1...d_{n_d}]$$

(4.3a)

$$\Phi^T(t+1) = [-\bar{y}(t)...-\bar{y}(t-n_a+1)\bar{u}(t)...\bar{u}(t-n_b+1)$$

$$- w(t)...-w(t-n_f+1)\epsilon(t)...\epsilon(t-n_c+1)$$

$$- \gamma(t)...-\gamma(t-n_d+1)]$$

(4.3b)

For the difference equation (2.2), The RPEM is given by the following recursive equations:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + X(t)\epsilon(t)$$

(4.4a)

$$X(t) = \frac{P(t-1)\Psi(t)}{\lambda(t) + \Psi^T(t)P(t-1)\Psi(t)}$$

(4.4b)

$$P(t) = P(t-1)\left[I - \frac{\Psi(t)\Psi^T(t)P(t-1)}{\lambda(t) + \Psi^T(t)P(t-1)\Psi(t)}\right]\frac{1}{\lambda(t)}$$

(4.4c)

where $\lambda(t)$ is the forgetting factor and $\Psi(t)$ is the gradient of the prediction error. If we introduce the following variables:

$$v(t) = \frac{D(q^{-1})}{C(q^{-1})}\bar{y}(t) \quad (4.5a) \qquad \beta(t) = \frac{D(q^{-1})}{C(q^{-1})F(q^{-1})}\bar{u}(t) \quad (4.5b)$$

$$\rho(t) = \frac{1}{C(q^{-1})}\epsilon(t) \quad (4.5c) \qquad \sigma(t) = \frac{1}{C(q^{-1})}\tau(t) \qquad (4.5d)$$

$$\delta(t) = \frac{D(q^{-1})}{C(q^{-1})F(q^{-1})}w(t) \quad (4.5e)$$

$\Psi(t)$ can be written as

$$\Phi^T(t+1) = [-v(t)...-v(t-n_a+1)\beta(t)...\beta(t-n_b+1)$$

$$- \delta(t)...-\delta(t-n_f+1)\rho(t)...\rho(t-n_c+1)$$

$$- \sigma(t)...-\sigma(t-n_d+1)]$$

(4.6)

If $\Psi(t)$ is replaced by $\Phi(t)$ the corresponding algorithm is termed as a Recursive PseudoLinear Regressions (RPLR).

If we modify the equations (4.5) as

$$\bar{v}(t) = \frac{D(kq^{-1})}{C(kq^{-1})}\bar{y}(t) \qquad \bar{\beta}(t) = \frac{D(kq^{-1})}{C(kq^{-1})F(kq^{-1})}\bar{u}(t)$$

$$\bar{\rho}(t) = \frac{1}{C(kq^{-1})}\epsilon(t) \qquad \bar{\sigma}(t) = \frac{1}{C(kq^{-1})}\tau(t)$$

$$\bar{\delta}(t) = \frac{D(kq^{-1})}{C(kq^{-1})F(kq^{-1})}w(t)$$

where $0 < k < 1$. We obtain a more general algorithm that reduces to the RPEM (k=1) and the RPLR (k=0) as special cases. The " contraction factor " k can be variable as Friedlander (1982) suggested:

$$k(t) = \rho k(t-1) + (1-\rho) \qquad k(0) = 0$$

where ρ is a constant somewhat smaller than 1, e.g. 0.98-0.99. In this paper we choose $\rho = 0.99$.

5 Test Cases for Comparison and Evaluation

The discrete-time transfer functions of the simulated processes are given by two third order low pass processes:

$$G_1(q^{-1}) = \frac{0.065q^{-1} + 0.048q^{-2} - 0.008q^{-3}}{1 - 1.500q^{-1} + 0.705q^{-2} - 0.100q^{-3}} \quad (5.1)$$

$$G_2(q^{-1}) = \frac{0.0186q^{-1} + 0.0486q^{-2} + 0.0078q^{-3}}{1 - 1.7063^{-1} + 0.9580q^{-2} - 0.1767q^{-3}}$$

$$(5.2)$$

one second order low-pass processes:

$$G_3(q^{-1}) = \frac{0.1387q^{-1} + 0.0889q^{-2}}{1 - 1.036q^{-1} + 0.2636q^{-2}} \qquad (5.3)$$

one oscillating unstable process:

$$G_4(q^{-1}) = \frac{0.1964q^{-1} + 0.0001q^{-2} - 0.1892q^{-1}}{1 - 2.93q^{-1} + 2.866q^{-2} - 0.9277q^{-3}}(5.4)$$

where $G_2(q^{-1})$ has one zero outside of the unit circle of the z-plane. The output of the processes is contaminated by an autocorrelated discrete noise n(t) which is generated by an ARMA model

$$n(t) = \tau\frac{1 + 0.500q^{-1} + 0.250^{-2}}{1 - 0.527q^{-1} + 0.0695q^{-2}}e(t) \qquad (5.5)$$

or

$$n(t) = \tau\frac{1 + 0.05q^{-1} + 0.80q^{-2}}{1 + 1.0q^{-1} + 0.20q^{-2}}e(t) \qquad (5.6)$$

where e(t) is a normally distributed white noise sequence with zero mean and unity standard deviation. The scalar weighting factor, τ , is used to adjust the noise-to-signal ratio.

6 Comparison and Evaluation of Linear Model Structures

In this section we shall discuss and compare the self-tuning control results of the processes (4.1-4.4), in which the linear model sets (2.3-2.10) have been used in controller design respectively. For the reason of space limitation, we only give and discuss the simulation results of the model: LM(A,B), LM(A,B,C), and LM(A,B,D), which are most commonly in STAC.

In each simulation case, the number of the parameters to be estimated was the same for each model structure and the parameter estimates were initialized to zero, except for unit $b_0(0)$. The initial covariance matrix was taken as 10 times the unit matrix. Default settings of the various tuning knobs were chosen with $K_{min}=1$, $N_1=K_{min}$, P=1, R=1, $N_2=4$ and $N_U=4$ for the GPC method. The control weighting factor, μ, was set to 0.05 with the exception of the MV controller. Furthermore, the control signal and the plant output were limited to lie in the range [-10, +10] and a reproducible noise sequence e(t) was generated and used in all cases.
The measures of model validity were taken as the variance of output error

$$\sigma_0^2 = \frac{1}{M}\sum_{t=1}^{M}(y(t) - w(t))^2 \qquad (6.1)$$

and synthetic measure

$$J_p = \frac{1}{M}\sum_{t=1}^{M}[(y(t) - w(t))^2 + \mu(u(t))^2] \qquad (6.2)$$

where M is the number of samples, μ the control weighting factor.

Simulation Results of The Third Order Low-Pass Process

First the simulation results of the third order low-pass processes (5.1) with the noise model (5.5) are discussed. The various linear model sets were used in the GMV controller design, respectively, for the simulated process. The noise weighting factor was taken as 0.1 and 0.05, respectively. For each noise weighting factor, the four cases were regarded as follows:

(a) the RPEM with variable forgetting factor,
(b) the RPLR with variable forgetting factor,
(c) the RPEM with constant forgetting factor of 0.95, and
(d) both forgetting factor and contraction factor are variable.

The results are shown in Table 6.1. The underlined values show the cases in which the performance of the closed loop system was extremely degraded. Figs. 6.1-6.3 show the input-output behaviour of LM(A,B), LM(A,B,C), and LM(A,B,D) for the case (d).

Table 6.1 the numerical results of the low-pass process (4.1) with the GMV self-tuner

(a)   M=480, and τ=0.10

Model Sets	n_a	n_b	n_c	n_d	case (a)		case (b)	
					J_p	σ_o	J_p	σ_o
LM(A,B)	4	4			0.621	0.326	0.621	0.326
LM(A,B,C)	3	3	2		2.836 (0.627)	2.301 (0.347)	1.564 (0.628)	1.054 (0.326)
LM(A,B,D)	3	3		2	0.674	0.390	0.627	0.325
					case (d)		case (c)	
LM(A,B)	4	4			0.621	0.326	0.808	0.483
LM(A,B,C)	3	3	2		3.181 (0.628)	2.620 (0.335)	1.112 (4.793)	0.563 (4.280)
LM(A,B,D)	3	3		2	0.626	0.328	1.000	0.660

(b)   M=500, and τ=0.05

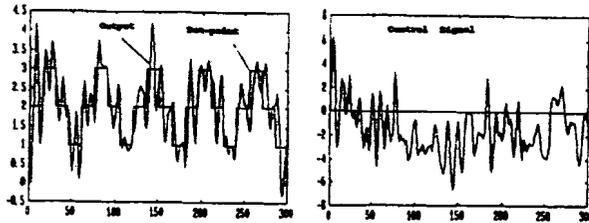| Model Sets	n_a	n_b	n_c	n_d	case (a)		case (b)	
					J_p	σ_o	J_p	σ_o
LM(A,B)	4	4			0.364	0.258	0.364	0.258
LM(A,B,C)	3	3	2		0.378	0.272	0.368	0.262
LM(A,B,D)	3	3		2	0.369	0.265	0.363	0.258
					case (d)		case (c)	
LM(A,B)	4	4			0.364	0.258	0.397	0.285
LM(A,B,C)	3	3	2		0.362	0.255	0.427	0.313
LM(A,B,D)	3	3		2	0.368	0.260	0.365	0.258

Fig. 6.1 The input-output behaviour of the closed loop system for the LM(A,B) with the case (d)
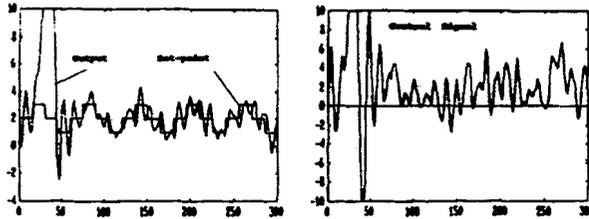

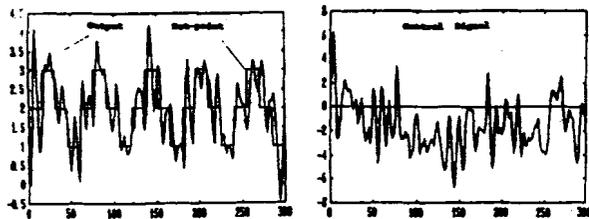**Fig. 6.2 The input-output behaviour of the closed loop system for the LM(A,B,C) with the case (d)**

Fig. 6.3 The input-output behaviour of the closed loop system for the LM(A,B,D) with the case (d)

Simulation Results of Other Processes

The numerical results of the second order low-pass process (5.3) with the noise model (5.6) for the MV method are shown in Table 6.2. In this case, $\sigma_0^2 = J_p$ in (6.1) and (6.2).

Tables 6.3 and 6.4 show the results of the third order non-minimum phase process (5.2) and the third order oscillating unstable process (5.4) with the noise model (5.5), in which the GPC controller was used for evaluation.

Table 6.5 show the numerical results of a time-varying process, in which every 60 samples the simulated process was switched between the process (5.1) and the third order process

$$G(q^{-1}) = \frac{0.065q^{-1} + 0.048q^{-2} - 0.008q^{-3}}{1 - 1.7063q^{-1} + 0.958q^{-2} - 0.1767q^{-3}}$$

and the noise model was switched between (5.5) and (5.6).

Table 6.2 The numerical results of the low-pass process (4.3) with the MV self-tuner   M=480   and τ =0.10
λ(t)=0.99λ(t-1)+0.01   λ(0)=0.95

Model Sets	n_a	n_b	n_c	n_d	RPLR	RPEM
LM(A,B)	3	3			0.033	0.033
LM(A,B,C)	2	2	2		5.434 (0.043)	0.033
LM(A,B,D)	2	2		2	0.030	0.027

Table 6.3 The numerical results of the nonminimum phase process (4.2) with the GPC self-tuner   M=500, and τ =0.10
λ(t)=0.99λ(t-1)+0.01   λ(0)=0.95

Model Sets	n_a	n_b	n_c	n_d	RPEM		RPLR	
					J_p	α	J_p	α
LM(A,B)	4	4			0.634	0.193	0.634	0.193
LM(A,B,C)	3	3	2		0.687	0.221	6.396 (0.865)	4.121 (0.226)
LM(A,B,D)	3	3		2	0.904	0.217	0.871	0.219

Table 6.4 The numerical results of the third order unstable process (4.2) with the GPC self-tuner
M=500, τ =0.10, and $N_2=N_u=4$
λ(t)=0.99λ(t-1)+0.01   λ(0)=0.95

Model Sets	n_a	n_b	n_c	n_d	RPEM		RPLR	
					J_p	α	J_p	α
LM(A,B)	4	4			0.481	0.384	0.481	0.384
LM(A,B,C)	3	3	2		1.185	0.320	1.207	0.324
LM(A,B,D)	3	3		2	1.089	0.320	1.101	0.321

Table 6.5 The numerical results of the time-varying process with the GMV self-tuner   M=240, and τ =0.05
λ(t)=0.99λ(t-1)+0.01   λ(0)=0.95

Model Sets	n_a	n_b	n_c	n_d	RPEM		RPLR	
					J_p	α	J_p	α
LM(A,B)	4	4			0.741	0.562	0.741	0.562
LM(A,B,C)	3	3	2		0.501	0.366	13.36 (2.209)	12.14 (1.514)
LM(A,B,D)	3	3		2	0.581	0.441	1.209	0.915

Simulation results can be summarized as follows:

. The best control performance is provided either by LM(A,B,D) or LM(A,B) in most cases.

. LM(A,B,D) and LM(A,B) are robust to the variation of noise level, forgetting factor, and contraction factor, while

2204

LM(A,B,C) is very sensitive.
. LM(A,B,C) perform poorly in many cases.

For LM(A,B,C), the problem arises from the noise polynomial $C(q^{-1})$ with the roots outside the disc or on the unit circle. From the prefilters in the (4.5) and the closed loop characteristic equation (3.15), it follows that an important assumption for LM(A,B,C) is that the roots of the polynomial $C(q^{-1})$ must lie inside the unit circle. When the polynomial $C(q^{-1})$ does not satisfy the assumption, the closed loop system either performs poorly or becomes unstable.

According to the spectral factorization theorem, this problem can theoretically be avoided by monitoring the stability of the polynomial $C(q^{-1})$, and replacing the unstable roots of the polynomial $C(q^{-1})$ by its reciprocal. In practice, the performance can not sometimes be improved via replacement of the unstable roots.

The numerical results with stability monitoring and replacement of roots are listed in brackets of Table 6.1 for the process (5.1). Due to the replacement of the unstable roots in the polynomial $C(q^{-1})$, the performance of closed-loop system even becomes worse for the RPEM with the constant forgetting factor of 0.95 even through the improvement is obvious in other cases.

In general, the foregoing discussions on LM(A,B,C) are also suitable to other model sets with the polynomial $C(q^{-1})$.

On Computational Expense

The number of computations on the parameter estimation part have been given by [3] for the case without stability monitoring. For the RPEM, the least computational expense is given by LM(A,B), and the next by LM(A,B,D), and then LM(A,B,C).

For the controller design part, the calculation related to the model structure includes resursion of the Diophantine equations (3.2) and (3.3) and synthesis of the controller (3.10). Table 6.6 lists the computation number of solving recursively (3.2) and (3.3) and synthesizing the GPC controller for the three type of model sets, where the degree of polynomials was assumed as $n=n_a=n_b=n_c=n_d$. The least computation number is given by LM(A,B), and the second by LM(A,B,C), and then LM(A,B,D).

When the stability monitoring is performed on-line, LM(A,B,C) need the additional computational burden and become time-consuming, especially for the polynomial $C(q^{-1})$ with high degree.

7 Conclusions

The simulation results have shown that LM(A,B,C) is not a model as one expects, which requires the additional stability monitoring and are sensitive to the

variation of the identification algorithm and the forgetting factor.

LM(A,B) and LM(A,B,D) seem to have the most advantages: very good performance and overall reliability for various types of processes. LM(A,B) is also advantageous for the least computational expense and simplicity.

Table 6.6 The total computation number of solving the Diophantine equations and synthesizing the controller for the three type of linear model sets with the assumed model orders

Model Sets	Multiplications	Additions
LM(A,B)	$N_2(6n+2)-3n/2-2$	$N_2(6n+2)-2$
LM(A,B,C)	$N_2(6n+2)-2$	$N_2(6n+1)+2n-2$
LM(A,B,D)	$N_2(8n+2)+n^2-n-2$	$N_2(8n+2)+2n^2-4n-2$

References

[1] D. W. Clark, C. Mohtadi, and P. S. Tuffs,"Generalized Predictive control Part 1. the basic algorithm", Automatica. Vol.23, No.2, pp 149-160, 1987

[2] D. W. Clark, C. Mohtadi, and P. S. Tuffs,"Generalized Predictive control Part 2. extensions and interpretations", Automatica. Vol.23, No.2, pp.149-160, 1987

[3] F. Conrad and J. J. Zhou,"On-line Identification of a hydraulic servo system", Proc. IFAC-Symposium on Identification and Parameter Estimation, Beijing, August, pp.828-836, 1988

[4] B. Friedlander,"A Modified Prefilter for Some Recursive Parameter Estimation Algorithms", IEEE Transactions on Automatic Control, Vol. AC-27, No.1, pp. 232-235, 1982

[5] A. M. Lelic, and B. M. Zarrop,"Generalized Pole-Placement Self-Tuning Controller Part 1. Basic Algorithm", Int. J. Control, Vol.46, No.2, pp.547-568, 1987

[6] A. M. Lelic, and B. M. Zarrop, "Generalized Pole-Placement Self-Tuning Controller, Part 2. Application to robot manipulator control", Int. J. Control, Vol.46, No.2, pp.569-601, 1987

[7] L. Ljung and T. Soderstrom, "Theory and Practice of Recursive Identification", MIT Press, Cambridge, Mass. 1983

[8] P. S. Tuffs and W. D. Clark, "Self-tuning control of offset: a unified approach", IEE Proc. Vol. 132, Pt. D, No. 3, 100-110. 1984