第7卷 第3期 控制理论与应用 Vol. 7, No. 3
1990年9月 CONTROL THEORY AND APPLICATIONS Sept., 1990

A Recurstve Generalized Predictive Control Algorithm for ARMAX Models*

Xu Lihong
(Institute of Automation, Southeast University, Nanjing)
Yuan Zhendong
(Department of Mathematics, East China Normal University, Shanghai)

Abstract: A recursive generalized predictive control (RGPC) algorithm for ARMAX models is proposed. This algorithm needs less computational effort than GPC(D. W. Clarke (1987)) and is also better than RGPSTC(Yuan, Z. Z. (1989)) because it can be used in the case of singular matrices which are contained in the GPC control law. The simulation study shows that the RGPC is robust.

Key words: adptive control; predictive control; self—tuning control; recursive algorithm; least—squares estimate

1. Introduction

The generalized predictive control (GPC) (D. W. Clarke et al (1987)) is robust but needs great computational effort. To solve the problem, Yuan Z. Z. (1989) proposed a recursive GPSTC. However this algorithm is not suitable for the case of singular matrix which are contained in GPC control law.

To overcome the above difficulty, we will develope a recursive generalized predictive control (RGPC) algorithm for the ARMAX process in this paper.

In section 2, the performance index is given. We give the main results in section 3 and section 4. The recursive predictive control algorithm is completely given in section 5 and some simulation studies are stated in section 6. The conclusion is given in section 7.

2. The Performance Index

Consider the following ARMAX model

$$A(q^{-1})y(t) = B(q^{-1})u(t-1) + C(q^{-1})e(t), \tag{2.1}$$

where e(t) is the zero mean white noise, A, B and C are polynomials in the q^{-1}:

$$A(q^{-1}) = 1 + a_1q^{-1} + \ldots + a_{n_a}q^{-n_a};\ B(q^{-1}) = b_0 + b_1q^{-1} + \ldots + b_{n_b}q^{-n_b};$$

$$C(q^{-1}) = 1 + c_1q^{-1} + \ldots + c_{n_c}q^{-n_c}.$$

* This work was supported by National Science Foundation of China.
Manuscript received Mar. 27, 1989, revised Oct. 21, 1989.

For simplicity, we assume that $n_a = n_c = n$.

Performance index

$$J = E\{\sum_{h=1}^{N}[y(t+h) - y_r(t+h)]^2 + \sum_{h=1}^{M}\lambda(h)u^2(t+h-1)\}, \quad (2.2)$$

where $\quad y_r(t+h) = \alpha^h y(t) + (1-\alpha^h)w(t), (0<\alpha<1) \quad (2.3)$

$w(t)$ is the set-point. So, $y_r(t+h)$ is the smoothed set-point. N, M are the prediction and control horizon respectively ($M \leqslant N$), $\lambda(h)$ is a control weighting sequence. In general, we assume $\lambda(h) = \lambda > 0$, λ is a constant.

3. The Predictive Control Law

According to the Diophantine equation:

$$C = AF_h + q^{-h}G_h \qquad (\text{where } \deg F_h = h-1, F_h(0) = 1, \deg G_h = n-1). \quad (3.1)$$

From (2.1) and (3.1), we obtain

$$y(t+h) = y^*(t+h|t) + \xi(t+h), \quad (3.2)$$

where

$$y^*(t+h|t) = \frac{G_h}{C}y(t) + \frac{BF_h}{C}u(t+h-1) \quad (3.3)$$

$$\xi(t+h) = F_h e(t+h) \quad (3.4)$$

Since $\xi(t+h)$ is independent of $y^*(t+h|t)$, $y^*(t+h|t)$ is the h-step ahead prediction of $y(t)$.

Hence

$$J = J_1 + \delta^2, \quad (3.5)$$

where

$$J_1 = \sum_{h=1}^{N}[y^*(t+h|t) - y_r(t+h)]^2 + \lambda\sum_{h=1}^{M}u^2(t+h-1); \quad \delta^2 = \sum_{h=1}^{N}E\xi^2(t+h). \quad (3.6)$$

Minimizing J with respect to U is equal to minimizing J_1.

Let

$$X(q^{-1}) = X_{k-1}(q^{-1}) + q^{-k}X_{k-1}^*(q^{-1}), \quad (3.7)$$

where

$$X_{k-1}(q^{-1}) = 1 + x_1q^{-1} + \dots + x_{k-1}q^{-k+1};$$

$$X_{k-1}^*(q^{-1}) = \begin{cases} x_k + x_{k+1}q^{-1} + \dots + x_{n_x}q^{-n_x+k} & as\ k \leqslant n_x \\ 0 & as\ k > n_x \end{cases}$$

(In this paper, X represent A, B, C and x represent a, b, c respectively) Henceforth we assume that

$$a_k = c_k = 0 \text{ as } k > n, \quad b_k = 0 \text{ as } k > n_b. \quad (3.8)$$

From (2.1), (3.7) and (3.2), we have

$$\begin{aligned} y(t+k) = &- a_1y^*(t+k-1|t) - \dots - a_{k-1}y^*(t+1|t) - A_{k-1}^*y(t) \\ &+ B_{k-1}u(t+k-1) + B_{k-1}^*u(t-1) + C_{k-1}^*e(t) \\ &+ [C_{k-1}e(t+k) - a_1\xi(t+k-1) - \dots - a_{k-1}\xi(t+1)]. \end{aligned} \quad (3.9)$$

with (3.4), the terms in brackets of (3.9) are independent of $e(t), e(t-1), \dots, e(0)$. Hence the k-step ahead prediction of $y(t)$ is

$$\begin{cases} y^*(t+k|t) = -a_1y^*(t+k-1|t) - \ldots - a_{k-1}y^*(t+1|t) \\ \qquad - A_{k-1}^*y(t) + B_{k-1}u(t+k-1) + B_{k-1}^*u(t-1) + C_{k-1}^*e(t), \\ e(t) = -C_0^*e(t-1) + Ay(t) - Bu(t-1). \end{cases} \tag{3.10}$$

(3.11)

Let $k=1,2,\ldots,N$ respectively, according to (3.10) we can obtain the following equation:

$$y^*(t+k|t) = S_{k-1}u(t+k-1) + \tilde{B}_{k-1}^*u(t-1) - \tilde{A}_{k-1}^*y(t) + \tilde{C}_{k-1}^*e(t) \tag{3.12}$$

and the following recursive relationships:

$$S_{k-1} = B_{k-1} - \sum_{i=1}^{k-1} a_iS_{k-1-i}q^{-i}, \qquad S_0 = B_0 = b_0, \tag{3.13}$$

$$\tilde{X}_{k-1}^* = X_{k-1}^* - \sum_{i=1}^{k-1} a_i\tilde{X}_{k-1-i}^*, \qquad \tilde{X}_0^* = X_0^*. \tag{3.14}$$

From (3.13) we can obtain: $\quad S_k = S_{k-1} + s_kq^{-k},$ (3.15)

hence we can denote the polynomials $S_{k-1}, \tilde{X}_{k-1}^*$ in the following forms:

$$S_{k-1} = s_0 + s_1q^{-1} + \ldots + s_{k-1}q^{-k+1}, \tag{3.16}$$

$$\tilde{X}_{k-1}^* = \tilde{X}_{k-1,1}^* + \tilde{X}_{k-1,2}^*q^{-1} + \ldots + \tilde{x}_{k-1,n_x}^*q^{-n_x+1}. \tag{3.17}$$

We asume $\quad u(t+k-1) = u(t+M-1) \quad$ as $M<k\leqslant N$. (3.18)

then (3.12) can be written in vector form:

$$Y^* = GU + f, \tag{3.19}$$

Where

$$Y^* = [y^*(t+1|t), y^*(t+2|t), \ldots, y^*(t+N|t)]^\tau, U = [u(t), u(t+1), \ldots, u(t+M-1)]^\tau,$$

$$G = \begin{bmatrix} s_0 & & & 0 \\ \vdots & \ddots & & \\ s_{M-1} & \cdots & s_1 & s_0 \\ s_M & \cdots & s_2 & s_1+s_0 \\ \vdots & & & \vdots \\ s_{N-1} & \cdots & s_{N-M+1} & \sum_{i=0}^{N-M} s_i \end{bmatrix}_{N\times M}, \quad f = \begin{bmatrix} \tilde{B}_0^*u(t-1) - \tilde{A}_0^*y(t) + \tilde{C}_0^*e(t) \\ \tilde{B}_1^*u(t-1) - \tilde{A}_1^*y(t) + \tilde{C}_1^*e(t) \\ \cdots\cdots\cdots \\ \tilde{B}_{N-1}^*u(t-1) - \tilde{A}_{N-1}^*y(t) + \tilde{C}_{N-1}^*e(t) \end{bmatrix}. \tag{3.20}$$

Let

$$W = [y_r(t+1),\ y_r(t+2), \ldots,\ y_r(t+N)]^\tau \tag{3.21}$$

With (3.19) and (3.21), the (3.6) can be written as

$$J_1 = (GU + f - W)^\tau(GU + f - W) + \lambda U^\tau U. \tag{3.22}$$

So, $\quad U = (G^\tau G + \lambda I)^{-1}.G^\tau(W-f)$, from $\partial J_1/\partial U = 0$. (3.23)

So, $\quad u(t) = g^\tau(W-f)$, (3.24)

where g^τ is the first row of the matrix $(G^\tau G+\lambda I)^{-1}G^\tau$.

According to (3.13) and (3.14), with mathematical induction, the recursions of $s_m, \tilde{x}_{m,i}^*$ are given by

$$s_m = b_m - \sum_{j=1}^{m} a_j s_{m-j}, \quad s_0 = b_0, \tag{3.25}$$

$$\tilde{x}^*_{m,i} = x_{m+i} - \sum_{j=1}^{m} a_j \tilde{x}^*_{m-j,i}, \quad \tilde{x}^*_{0,i} = x_i, \tag{3.26}$$

$(m=1,2,\ldots,N-1, \quad i=\max(n_a,n_b,n_c))$

Remark We can now compute recursively all parameters of (3. 24) directly from the coefficients of A,B,C to avoid solving the Diophantine equation many times and, calculate all $\tilde{a}^*_{m,i}, \tilde{b}^*_{m,i}, \tilde{c}^*_{m,i}$ using the same computer programme ((3. 26)). So, this algorithm needs less computational effort than GPC and RGPSTC.

4. Recursively Computing $(G^\tau G+\lambda I)^{-1}$

Let $\quad H = G^\tau G + \lambda I \quad (4.1)$

and denote $\quad H^\tau = [h_1^\tau, \ldots, h_N^\tau]$, where h_i is the ith row of H, $\quad i=1,2,\ldots,N \quad (4.2)$

and let $\quad H_1 = h_1, \quad H_2 = \begin{bmatrix} H_1 \\ h_2 \end{bmatrix}, \ldots, \quad H_N = \begin{bmatrix} H_{N-1} \\ h_N \end{bmatrix}$, then $H = H_N$.

The computing blockdiagram of H^+ can be depicted as follows:

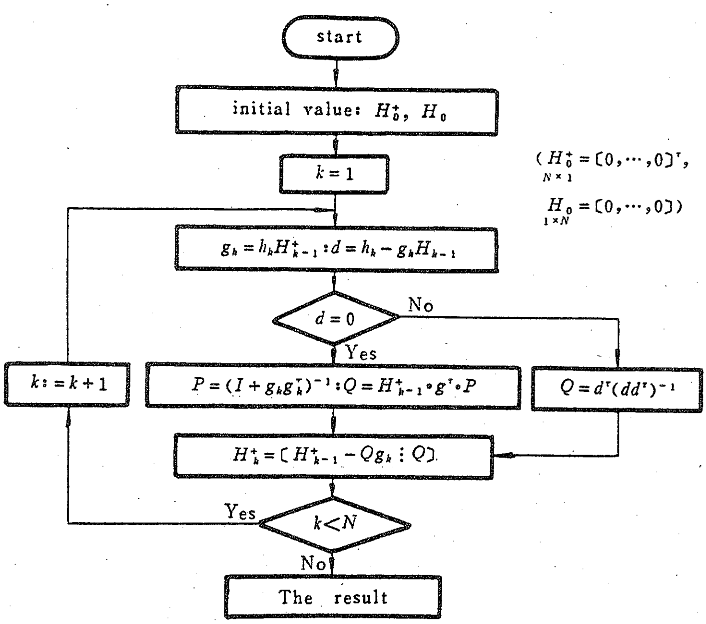

Fig. 1 The computing blockdiagram of H^+

Remark H^+ satisfies $H H^+ H = H$ and be called the generalized inverse of H. If the inverse matrix of H exists then $H^+ = H^{-1}$.

5. Recursive Generalized Predictive Control Algorithm (RGPC)

In the adaptive case, the coefficients of polynomials A, B, C are unknow. We can state the RGPC algorithm as follows:

Step 1 select the design parameters N, M, λ in (2.2) and α in (2.3).

Step 2 Estimate the unknow parameters θ using weighted least squares method:

$$\theta = [a_1, \ldots, a_{n_a}, \quad b_0, \ldots, b_{n_b}, \quad c_1, \ldots, c_{n_c}]^{\tau}.$$

The forgetting factor is β, at the instant k.

Step 3 Calculate s_m, $\tilde{b}^*_{m,i}$, $\tilde{a}^*_{m,i}$, $\tilde{C}^*_{m,i}$ with (3.25)—(3.26) to obtain $H = (G^{\tau}G + \lambda I)$.

Step 4 According to the computing brockdiagram Fig. (4.3) calculate H^+ and the first row g^{τ} of $H^+ \cdot G^{\tau}$.

Step 5 Calculate $u(t+1)$ from (3.24), then let $k := k+1$ and back to step 2.

6. Numerical Simulation

In simulations, we used the ARMAX model with $n_a = n_c = 2, n_b = 5$, During the first 80 samples the control law was acting on simulated system (1). During the second 80 samples the control law acting on simulated system (2) and so on.

The simulated systems are given in the following table:

(1) $y(t) - 1.756726y(t-1) + 0.778801y(t-2) = 1.011497u(t-1) + 0.010577u(t-2) + 0.1e(t)$

(2) $y(t) - 1.756726y(t-1) + 0.778801y(t-2) = 1.001097u(t-3) + 0.015921u(t-4) + 0.005056u(t-5) + 0.1e(t)$

(3) $y(t) - 0.904837y(t-1) = 1.029554u(t-3) + 0.065609u(t-4) + 0.1e(t)$

(4) $y(t) - 0.904837y(t-1) = 1.095163u(t-1) + 0.1e(t)$

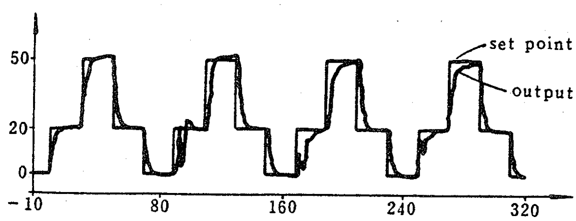

Fig. 2 The output of the systems using RGPC

The simulation shows that the RGPC is robust to the changing of order and the variant time delay (Fig. 1).

7. Conclutions

A recursive generalized predictive control (RGPC) algorithm has been derived based on the ARMAX model. This algorithm can compute recursively all parameters of the controller directly from the coefficients of the model and needs less computational effort than the GPC and suits the singular case of $(G^{\tau}G+\lambda I)$.

A series of simulation shows that the RGPC is robust enough.

Reference

[1] Clarke, D. W., C. Mohtadi & P. S. Tuffs, Generalized predictive control, Automatica, 23, 2, (1987), 137—162.

[2] Yuan, Z. Z., Recursive Generalized Predictive STC, ACTA. AUTOMATICA SINICA. 15, 4, (1989), 384—351.

ARMAX 模型的递推广义预测控制算法

徐 立 鸿

(东南大学自动化研究所，南京)

袁 震 东

(华东师范大学数学系，上海)

摘要 本文提出一种基于 ARMAX 模型的递推广义预测控制算法. 仿真研究表明这种算法对于系统阶和时滞变化等未建模动态是稳健的. 本算法计算量小于 Clarke 的 GPC 算法，并适用于 GPC 控制律中矩阵奇异的情况.

关键词： 自适应控制；预测控制；自校正控制；递推算法；最小二乘估计