

# 递推广义预测自校正控制器<sup>1)</sup>

袁 著 社  
(南开大学)

## 摘 要

本文给出了广义预测自校正控制器的递推算法,用递推平方根法代替最小二乘法估计参数,改善了计算精度,比牛津大学克拉克等人的算法节省了计算时间.仿真实验表明,该控制器对系统时延与阶次有很好的鲁棒性,性能优于PID调节器、广义最小方差自校正器与极点配置自校正器,有较强的抑制干扰能力.

**关键词**——预测控制,自校正控制,递推算法,鲁棒性.

## 一、引 言

十多年来产生了许多自校正器,都成功地用于实际过程.但是对变时延、变阶次与变参数过程,控制效果不好.因此,研制具有鲁棒性的自校正器成为人们关注的问题.

Richalet 等人<sup>[1]</sup>提出了大范围预测概念,在此基础上,Clarke 等人<sup>[2]</sup>提出了广义预测自校正器(GPC),但在线计算逆矩阵耗费时间长.本文提出的递推广义预测自校正器(RGPC)改进了GPC.仿真表明,RGPC有很好的鲁棒性,节省计算时间,用微机即可实现.

## 二、递推广义预测自校正控制器(RGPC)

**1. 预测模型** 用增量模型 CARIMA 描述具有非平稳噪声的工业过程

$$A(q^{-1})y(t) = B(q^{-1})u(t-1) + C(q^{-1})\xi(t)/\Delta, \quad (1)$$

其中

$$A(q^{-1}) = 1 + a_1q^{-1} + \dots + a_{n_a}q^{-n_a},$$

$$B(q^{-1}) = b_0 + b_1q^{-1} + \dots + b_{n_b}q^{-n_b},$$

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

$q^{-1}$  为向后移位算子,  $\{\xi(t)\}$  为不相关噪声.为了简单,令  $C(q^{-1}) = 1$ . 由模型(1)与丢番图方程

$$1 = E_j(q^{-1})\bar{A}(q^{-1}) + q^{-j}F_j(q^{-1}) \quad (2)$$

可得输出  $y(t+j)$  ( $1 \leq j \leq N$ ,  $N$  为最大预测范围)的最小方差预测值

本文于1986年11月4日收到.

1) 中国自然科学基金资助.

$$\hat{y}(t+j|t) = G_j \Delta u(t+j-1) + F_j y(t). \quad (3)$$

于是有

$$y(t+j) = G_j \Delta u(t+j-1) + F_j y(t) + E_j \xi(t+j). \quad (4)$$

其中  $\bar{A}(q^{-1}) = A(q^{-1})\Delta$ ,  $G_j = E_j B$ .

**2. 控制律** 为了使输出  $y(t)$  能够平稳地跟踪设定值  $y_r$ , 用一阶模型

$$\begin{aligned} w(t) &= y_r, \text{ 对 } j=1, 2, \dots, N, \text{ 有} \\ w(t+j) &= \alpha w(t+j-1) + (1-\alpha)y_r, \end{aligned} \quad (5)$$

产生新的设定值序列  $\{w(t+j), j=1, 2, \dots, N\}$ ,  $0 \leq \alpha \leq 1$ .

目标函数

$$J = E \left\{ \sum_{j=1}^N [y(t+j) - w(t+j)]^2 + \sum_{j=1}^N \lambda(j) [\Delta u(t+j-1)]^2 \right\}. \quad (6)$$

从  $j=1, 2, \dots, N$  用预测模型(4)得向量方程

$$\mathbf{y} = G\tilde{\mathbf{u}} + \mathbf{f} + \mathbf{e}. \quad (7)$$

其中  $\mathbf{y} = [y(t+1), \dots, y(t+N)]^T$ ,  $\tilde{\mathbf{u}} = [\Delta u(t), \dots, \Delta u(t+N-1)]^T$ ,

$\mathbf{f} = [f(t+1), \dots, f(t+N)]^T$ ,  $\mathbf{e} = [E_1 \xi(t+1), \dots, E_N \xi(t+N)]^T$ .

矩阵  $G$  为

$$G = \begin{bmatrix} g_0 & 0 & \dots & 0 \\ g_1 & g_0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \dots & 0 \\ g_{N-1} & g_{N-2} & \dots & g_0 \end{bmatrix}_{N \times N}. \quad (8)$$

$\mathbf{f}$  的分量  $f(t+1) = [G_1 - g_0] \Delta u(t) + F_1 y(t)$ ,  $f(t+2) = q[G_2 - q^{-1}g_1 - g_0] \Delta u(t) + F_2 y(t), \dots$ . (9)

以上公式中用到的  $E_j, F_j$  由公式(2)产生,  $G_j = E_j B, (j=1, 2, \dots, N)$ . 记

$$\mathbf{w} = [w(t+1), \dots, w(t+N)]^T,$$

则式(6)化为

$$J = E[(\mathbf{y} - \mathbf{w})^T(\mathbf{y} - \mathbf{w}) + \lambda \tilde{\mathbf{u}}^T \tilde{\mathbf{u}}]. \quad (10)$$

其中  $\lambda = \lambda(j)$ , 最小化  $J$  得预测控制律

$$\tilde{\mathbf{u}} = (G^T G + \lambda I)^{-1} G^T (\mathbf{w} - \mathbf{f}). \quad (11)$$

第一个分量

$$u(t) = u(t-1) + \bar{\mathbf{g}}^T (\mathbf{w} - \mathbf{f}). \quad (12)$$

其中  $\bar{\mathbf{g}}^T$  是矩阵  $(G^T G + \lambda I)^{-1} G^T$  的第一行.

为了计算简单, 指定一个控制范围,  $NU (< N)$ , 当  $j > NU$  时, 式(6)中有

$$\Delta u(t+j-1) = 0,$$

于是式(7)为

$$\mathbf{y} = \bar{G}\tilde{\mathbf{u}} + \mathbf{f} + \mathbf{e}. \quad (13)$$

其中

$$\bar{G} = \begin{bmatrix} g_0 & 0 & \cdots & 0 \\ g_1 & g_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & & g_0 \\ g_{N-1} & g_{N-2} & \cdots & g_{N-NU} \end{bmatrix}_{N \times NU}.$$

控制律变成

$$\bar{u} = [\bar{G}^T \bar{G} + \lambda I]^{-1} \bar{G}^T (w - f). \quad (14)$$

其中  $\bar{G}^T \bar{G} + \lambda I$  是一个  $NU \times NU$  对称矩阵.

### 3. 逆矩阵 $(\bar{G}^T \bar{G} + \lambda I)^{-1}$ 的递推算法<sup>[3]</sup>

令

$$\bar{G}^T \bar{G} + \lambda I = \left[ \begin{array}{c|c} A_{11}^{NU-1} & A_{12}^{NU-1} \\ \hline A_{21}^{NU-1} & A_{22}^{NU-1} \end{array} \right], \quad (15)$$

记  $M_{NU} = (\bar{G}^T \bar{G} + \lambda I)^{-1}$ , 可得递推公式:

置初值:

$$A_{11}^1, A_{12}^1, A_{21}^1, A_{22}^1, M_1 = \left( \lambda + \sum_{i=0}^{N-NU} g_i^2 \right)^{-1}.$$

从  $j = 1, 2, \dots, NU - 1$  计算

$$F_j = M_j A_{21}^j, D_j = [A_{11}^j - A_{12}^j F_j]^{-1}, L_j = F_j D_j,$$

从  $j = 1, 2, \dots, NU - 2$  计算

$$H_j = M_j + L_j F_j^T \text{ 与 } M_{j+1} = \left[ \begin{array}{c|c} D_j & -L_j^T \\ \hline -L_j & H_j \end{array} \right]. \quad (16)$$

从  $M_1$  开始, 由(16)式递推  $NU - 1$  步得  $M_{NU} = (\bar{G}^T \bar{G} + \lambda I)^{-1}$ .

### 4. RGPC 的计算步骤

- 1) 置初值:  $N, NU, n_a, n_b, \alpha, \beta$  与  $\lambda$ ;
- 2) 参数估计:  $A(q^{-1})\Delta y(t) = B(q^{-1})\Delta u(t-1) + \xi(t)$ ;
- 3) 递推求解丢番图方程:  $1 = E_j \bar{A} + q^{-i} F_j$ <sup>[2]</sup>;
- 4) 由(5)式计算设定值向量  $w$ ;
- 5) 由(9)式计算向量  $f$ ;
- 6) 由(16)式递推计算逆矩阵  $(\bar{G}^T \bar{G} + \lambda I)^{-1}$ ;
- 7) 计算矩阵  $(\bar{G}^T \bar{G} + \lambda I)^{-1} \bar{G}^T$  的第一行  $\bar{g}^T$ ;
- 8) 得 RGPC:  $u(t) = u(t-1) + \bar{g}^T (w - f)$ .

在每个采样区间上重复 2)–8)。

## 三、仿真实验结果

例 1. 过程模型(每 80 个采样区间变一次模型):

$$(1) \frac{1}{1 + 10s + 40s^2}, (2) \frac{e^{-2.7s}}{1 + 10s + 40s^2}, (3) \frac{e^{-2.7s}}{1 + 10s},$$



$$(4) \frac{1}{1+10s}, (5) \frac{1}{10s(1+2.5s)}$$

数字仿真时,采样间隔 1 秒,共 400 秒,输出设定值  $y_r$  在 0,20 与 50 之间变化,  $-10 \leq y(t) \leq 80$ ,  $-100 \leq u(t) \leq 100$ ,  $n_a = 2$ ,  $n_b = 6$ ,  $N = 6$ ,  $NU = 2$ ,  $\alpha = 0.6$ ,  $\lambda = 0.9$ . 仿真结果见图 1,表明系统对时延、阶次变化有很好的鲁棒性.

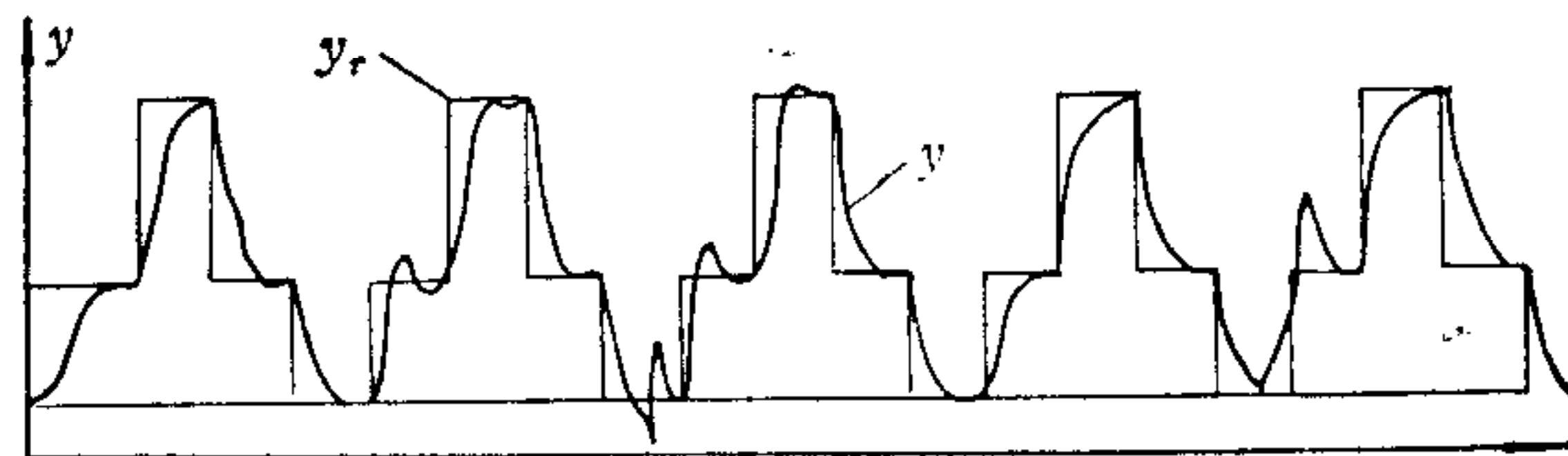


图 1 例 1 RGPC 的结果

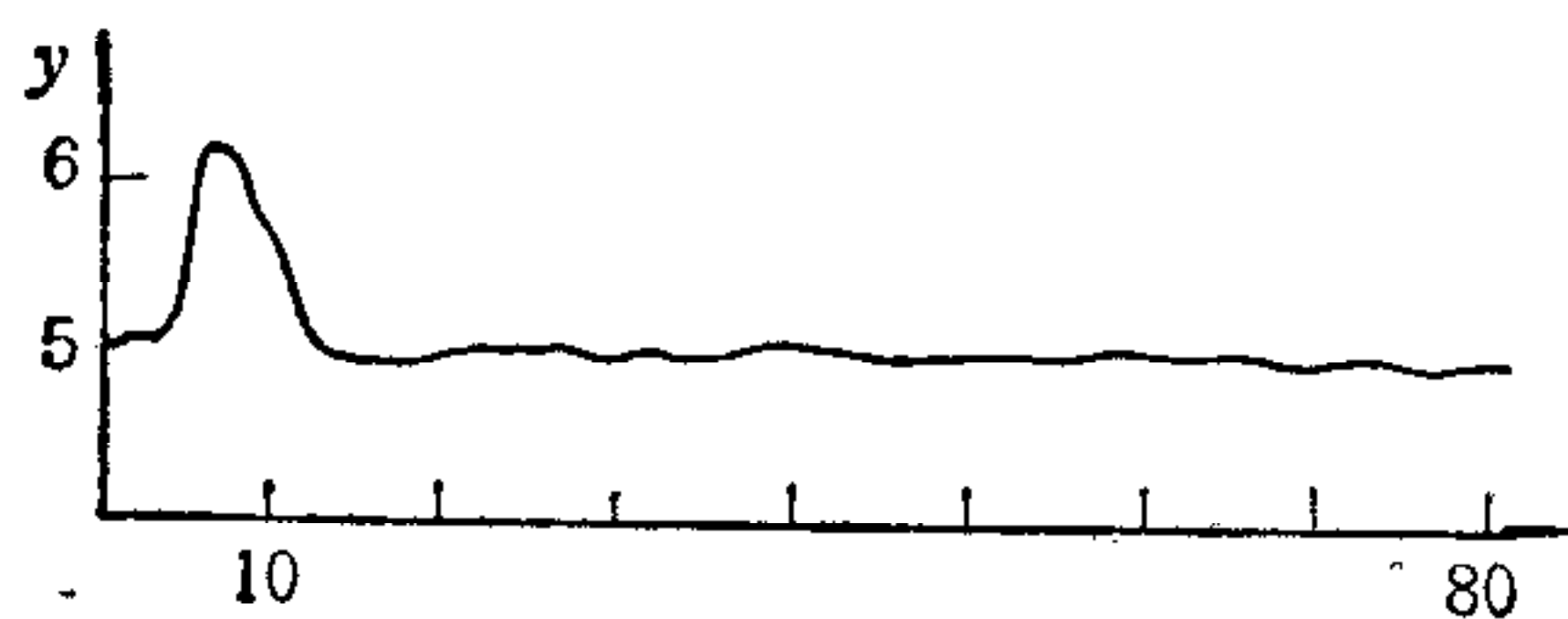


图 2 例 2 RGPC 的结果

## 例 2. 过程模型

$$y(t) = 0.99116y(t-1) - 0.06808u(t-3) + 0.06054v(t-3) + 0.02991\xi(t-1) + \xi(t).$$

其中负载干扰  $v(t) = 8.1$ , 随机干扰为  $\xi(t)$ . 控制器参数  $\alpha = 0.6$ ,  $N = 10$ ,  $NU = 2$ ,  $\lambda = 10$ ,  $n_a = 1$ ,  $n_b = 4$ ,  $y_r = 5$ . 用 RGPC 仿真,其结果见图 2,说明系统有很好的抗干扰能力,对负载变化没有静差.

## 参 考 文 献

- [1] Richalet, J. et al., Model Predictive Heuristic Control: Applications to Industrial Process, *Automatica*, 14 (1978), pp. 413—428.
- [2] Clarke, D. W. et al., Generalized Predictive Control, Part. 1, OUEL Report 1555/84(1984).
- [3] Yuan Zhuzhi and Liu Ruihua, RGPC-STC and Its Application to a Power Plant, Proc. of IEEE Asian Elec. Conf. (1987).

## RECURSIVE GENERALIZED PREDICTIVE STC

YUAN ZHUZHI  
(Nankai University)

### ABSTRACT

In this paper, a recursive algorithm for generalized Predictive STC(RGPC) is developed. The calculation needed is less than that of the GPC algorithm of Clarke et al. Using recursive squares root method instead of recursive least squares method. The RGPC has higher precision than GPC. Simulation shows that RGPC has a strong robustness to the change of dead-time and model order. It is superior to the accepted techniques such as PID, STC and Pole-placement. It has a strong robustness to disturbance.

**Key words** — Predictive control; self-tuning control; recursive algorithm; robustness.