# A New Predictive Self-Tuning Controller

MAO Zhizhong and YANG Lin

(Department of Automatic Control, Northeast University of Technology·Shengyang, 110006, PRC)

**Abstract**: By introducing a weighting polynomial in cost function, in this paper a self-tuning pole-placement controller based on long-range prediction of process output is proposed for a process which is described by process impulse response coefficients. The controller can ensure the stability of the control system and can eliminate the steady-state error of the control system, the computational load of the algorithm is less than other predictive control algorithm such as MAC, GPC and so on.

**Key words**: predictive control; pole-placement; self-tuning control; stability

## 1 Introduction

Much attention has been paid in recent years to self-tuning controller based on long-range predictive control methods[1,2]. The basic idea of these control algorithms can be described by following steps, at time $k$: i) The predictions of the process output from time $(k+1)$ to time $(k+N)$ are made based on a mathematical model of the process dynamics, the predictions are functions of control vector which consist of the future control actions from time $k$ to time $(k+N-1)$. ii) A control vector which minimizes a quadratic cost functions is calculated. iii) Only the first element of the control vector is applied to the process, at the next sampling period the whole procedure is repeated, that is, the receding horizon scheme is taken.

Richalet's MAC[2,3] (Model Algorithmic Control) method is one of the multistep receding horizon control method, it has some nice properties and has been implemented on a number of industrial processes. The success of the MAC operating on complex industrial processes is due, at least partially, to the impulse response representation of the processes. In fact, for most complex industrial processes, parametric models are difficult to obtain. It is known that parametric models can give results with large error if the order of the model does not agree with the order of the plant. Moreover in an industrial environment perturbations affect the plant structure more often than the measurable variable. This requires a constant checking and updating of model parameters. The impulse response representation is convenient, since in most industrial processes, the identification of the impulse response is relatively simple. However, when using MAC method, the transient response of control system is mainly regulated by a control weighting factor $\lambda$ and it is impossible to arbitrarily assign the poles of the control system by selecting the control weighting factor $\lambda$.

Manuscript received May. 24, 1992, revised Oct. 27, 1992.

It is significant to study the pole-placement controller based on long-range prediction control method, because this type of controller not only has nice properties of the long-range prediction controller, but also has nice properties of the pole-placement controller. Lelic[4] has been proposed a generalized pole-placement controller based on generalized predictive control[1], but the pole-placement controller based on non-parametric process models have been not reported yet. In this paper, a self-tuning pole-placement controller based on multistep cost function minimization is proposed for single-input single-output models described by process impulse response coefficients. The poles of the control system (that is, the poles of the controller) can be easily assigned, the computational load of the control algorithm is not larger than corresponding long-range predictive control algorithm in which the pole-placement technique is not taken. Theory analysis and simulation studies show that the control algorithm can eliminate the steady-state output tracking error and effectively rejects a load disturbance and stochastic disturbances.

## 2 A Self-Tuning Pole-Placement Controller

### 2.1 Process Model

Consider a linear time-invariant discrete-time process described as follows:

$$y(k) = \sum_{j=1}^{N} h_j u(k-j) = h(z^{-1})u(k), \tag{1}$$

where $y(k)$ and $u(k)$ are process output and input respectively, $h_j (j=1,2,\cdots N)$ are the coefficients of the process impulse response. As only a finite number of terms ($N$) is considered, the process is assumed to be stable and causal.

### 2.2 Predictor

When the process parameters are known, using eq. (1) a prediction $y(k+i/k)$, $i$ time-step to the future, can be written as:

$$\begin{aligned}
y(k+1/k) &= h_1 u(k) + h_2 u(k-1) + \cdots + h_N u(k-N+1), \\
y(k+2/k) &= h_1 u(k+1) + h_2 u(k) + \cdots + h_N u(k-N+2), \\
&\vdots \\
y(k+N-1/k) &= h_1 u(k+N-2) + h_2 u(k+N-3) + \cdots + h_N u(k-1).
\end{aligned}$$

The above equations can be written in the vector form:

$$Y(k) = HU(k) + \overline{H}\overline{U}(k), \tag{2}$$

where

$$\begin{aligned}
Y(k) &= [y(k+1/k)\ y(k+2/k)\ \cdots\ y(k+N-1/k)]^{\mathrm{T}}, \\
U(k) &= [u(k)\ u(k+1)\ \cdots\ u(k+N-2)]^{\mathrm{T}}, \\
\overline{U}(k) &= [u(k-N+1)\ u(k-N+2)\ \cdots\ u(k-1)]^{\mathrm{T}},
\end{aligned}$$

$$H = \begin{bmatrix} h_1 & 0 & 0 & \cdots & 0 \\ h_2 & h_1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ h_{N-1} & h_{N-2} & h_{N-3} & \cdots & h_1 \end{bmatrix}, \quad \overline{H} = \begin{bmatrix} h_N & h_{N-1} & h_{N-2} & \cdots & h_2 \\ 0 & h_N & h_{N-1} & \cdots & h_3 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & h_N \end{bmatrix}$$

Obviously, the process output's prediction $Y(k)$ is the function of the future control actions from time $k$ to time $(k+N-2)$.

### 2.3 Cost Function

Consider a cost function of the form:

$$J = \sum_{j=1}^{N-1}\{[y(k+j/k) - rw(k+j)]^2 + \lambda[Q_j(z^{-1})u(k+j-1)]^2\} \tag{3}$$

where $\{w(k)\}$ is a bounded reference sequence, $r$ is a quantity, $\lambda>0$ is a control weighting factor, $Q_j(z^{-1})$ $(j=1,2,\cdots,N-1)$ is defined as follows:

$$\begin{aligned} Q_1(z^{-1}) &= 1 + q_1z^{-1} + q_2z^{-2} + \cdots + q_{N-1}z^{-N+1}, \\ Q_2(z^{-1}) &= 1 + q_2z^{-2} + q_3z^{-3} + \cdots + q_{N-1}z^{-N+1}, \\ &\vdots \\ Q_{N-1}(z^{-1}) &= 1 + q_{N-1}z^{-N+1}. \end{aligned}$$

Now, define

$$\overline{Q} = \begin{bmatrix} q_{N-1} & q_{N-2} & q_{N-3} & \cdots & q_1 \\ 0 & q_{N-1} & q_{N-2} & \cdots & q_2 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & q_{N-1} \end{bmatrix}.$$

From Eq. (2) and above notation, the eq. (3) can be written as:

$$\begin{aligned} J =& [HU(k) + \overline{H}\overline{U}(k) - rW(k)]^{\mathrm{T}}[HU(k) + \overline{H}\overline{U}(k) - rW(k)] \\ &+ \lambda[U(k) + \overline{Q}\overline{U}(k)]^{\mathrm{T}}[U(k) + \overline{Q}\overline{U}(k)] \end{aligned} \tag{4}$$

where $W(k) = [w(k+1)\ w(k+2)\ \cdots\ w(k+N-1)]^{\mathrm{T}}$.

### 2.4 Control Algorithm

Assume no constraints on the future control actions, then the optimal control vector which minimize Eq. (4) is:

$$U(k) = (H^{\mathrm{T}}H + \lambda I)^{-1}[rH^{\mathrm{T}}W(k) - (H^{\mathrm{T}}\overline{H} + \lambda\overline{Q})\overline{U}(k)], \tag{5}$$

we will employ the receding horizon scheme, i. e. among the calculated optimal control actions, only the first one is applied to the process, and the whole procedure is repeated in the next sampling period, so the control law is:

$$u(k) = K^{\mathrm{T}}[rH^{\mathrm{T}}W(k) - (H^{\mathrm{T}}\overline{H} + \lambda\overline{Q})\overline{U}(k)], \tag{6}$$

where $K^{\mathrm{T}} = [1\ 0\ \cdots\ 0][H^{\mathrm{T}}H + \lambda I]^{-1} = [k_1\ k_2\ \cdots\ k_{N-1}]$.

The control law (6) generates the closed-loop equation:

$$P(z^{-1})y(k) = rh(z^{-1})K^{\mathrm{T}}H^{\mathrm{T}}W(k) \tag{7}$$

where $P(z^{-1}) = 1 + K^{\mathrm{T}}[H^{\mathrm{T}}\overline{H} + \lambda\overline{Q}][z^{-N+1}\ z^{-N+2}\cdots\ z^{-1}]^{\mathrm{T}}$ (8)

It is now required to eliminate the steady-state output tracking error and to arbitrarily assign the closed-loop poles by selecting the $r$ and $\overline{Q}$.

For correct tracking $r$ is chosen so that:

$$P(1) = rh(1)\overline{K}$$

that is: $r = P(1)/h(1)\overline{K}$ (9)

where $\overline{K} = K^{\mathrm{T}}H^{\mathrm{T}}[1\ 1\ \cdots\ 1]^{\mathrm{T}}$.

To assign the closed-loop poles select $\overline{Q}$ so that:

$$P(z^{-1}) = 1 + K^{\mathrm{T}}[H^{\mathrm{T}}\overline{H} + \lambda\overline{Q}][z^{-N+1}\ z^{-N+2}\cdots\ z^{-1}]^{\mathrm{T}} = T(z^{-1}) \tag{10}$$

where the zeros of $T(z^{-1})$ are the required poles of the control system, without loss generality, assume $T(0)=1$. Note $(H^{T}\overline{H})=(h_{ij})_{(N-1)\times(N-1)}$, then $(H^{T}\overline{H}+\lambda\,\overline{Q})$ can be written as:

$$H^{T}\overline{H}+\lambda\,\overline{Q}=\begin{bmatrix} h_{11}+\lambda q_{N-1} & h_{12}+\lambda q_{N-2} & \cdots & h_{1N-1}+\lambda q_{1} \\ 0 & h_{22}+\lambda q_{N-1} & \cdots & h_{2N-1}+\lambda q_{2} \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & h_{(N-1)(N-1)}+\lambda q_{N-1} \end{bmatrix},$$

so

$$1+[k_1 h_{1(N-1)}+\lambda q_1)+k_2(h_{2(N-1)}+\lambda q_2)+\cdots+k_{N-1}(h_{(N-1)(N-1)}+\lambda q_{N-1})]z^{-1}+\cdots$$
$$+[k_1(h_{12}+\lambda q_{N-2})+k_2(h_{22}+\lambda q_{N-1})]z^{-N+2}+k_1(h_{11}+\lambda q_{N-1})z^{-N+1}=T(z^{-1}). \quad (11)$$

The identity (11) contains $(N-1)$ linear equations for the coefficients $q_j (j=1,2,\cdots,N-1)$, obviously, if $k_1=0$, these equations can always be solved for arbitrarily given $T(z^{-1})$. In fact, $k_1$ is an element lying on first row and first array of matrix $(H^{T}H+\lambda I)^{-1}$, because $(H^{T}H+\lambda I)$ is a positive-definite matrix, it is not difficult to prove that $(H^{T}H+\lambda\, I)^{-1}$ is also a positive-definite matrix, so $k_1$ is always larger than zero.

If there is a load disturbance $d$ in the process (1), then the closed-loop equation (7) becomes:

$$P(z^{-1})y(k)=rh(z^{-1})K^{T}H^{T}W(k)+P(1)d.$$

In order to eliminate the effect of the load disturbance, the reference sequence $w(k)$ is replaced by $w(k)-w_0$, $w_0$ is chosen so that:

$$rh(1)\overline{K}w_0=P(1)d$$

that is:

$$w_0=d.$$

## 2.5 Self-Tuning Algorithm

When the process parameters in eq. (1) are unknown, a self-tuning control algorithm is formed by combining the above control algorithm with the parameter estimate algorithm:

1) Estimate the process parameters on-line using a recursive least squares algorithm:

$$\hat{\theta}(k)=\hat{\theta}(k-1)+M(k)\Phi(k)e(k), \quad (12)$$

$$M(k)=M(k-1)-\frac{M(k-1)\Phi(k)\Phi^{T}(k)M(k-1)}{1+\Phi^{T}(k)M(k-1)\Phi(k)} \quad (13)$$

where

$$e(k)=y(k)-\Phi^{T}\hat{\theta}(k-1),$$
$$\Phi(k)=[u(k-1)\ u(k-2)\ \cdots\ u(k-N)]^{T},$$
$$\hat{\theta}(k)=[\hat{h}_1(k)\ \hat{h}_2(k)\ \cdots\ \hat{h}_N(k)]^{T}.$$

2) Calculate the control action as follows:

$$u(k)=\frac{\hat{r}\hat{K}^{T}\hat{H}^{T}\overline{W}(k)}{P(z^{-1})} \quad (14)$$

where:

$$\hat{r}=P(1)/\hat{h}(1)\hat{\overline{K}}, \quad (15)$$
$$\overline{W}(k)=W(k)-[1\ 1\ \cdots\ 1]^{T}\hat{d}, \quad (16)$$

the unknown load disturbance $\hat{d}$ is estimated by following equation:

$$\hat{d}=y(k)-\Phi^{T}(k)\hat{\theta}(k). \quad (17)$$

## 2.6 Stability Analysis

About the stability of the self-tuning controller, we have following theorem:

**Theorem** When the self-tuning control algorithm (12)~(17) is applied to the process (1), the closed-loop system is BIBO stable and the steady-state error is zero.

**Proof** Substitution of eq. (15)~(17) into eq. (14) yields:

$$T(z^{-1})u(k) = \frac{T(1)}{\hat{h}(1)\bar{\hat{K}}}\hat{K}^{\mathrm{T}}W(k) - \frac{T(1)}{\hat{h}(1)}\varepsilon(k) \tag{18}$$

where

$$\varepsilon(k) = y(k) - \Phi^{\mathrm{T}}(k)\hat{\theta}(k). \tag{19}$$

Since $T(z^{-1})$ is a stable polynomial, $\{w(k)\}$ is a bounded sequence and $\hat{h}(1)\bar{\hat{K}} \neq 0$ ($\hat{h}(1)\bar{\hat{K}}$ is proportional to the steady-state gain of the control system), according to Lemma B. 3. 3 in [5], there exist constants $0 < m_1$, $m_2 < \infty$ so that:

$$|u(t)| < m_1 + m_2 \max_{1 \leqslant \tau \leqslant k} |\varepsilon(\tau)|, \quad 1 \leqslant t \leqslant k$$

and

$$\| \Phi(t) \| < N[m_1 + m_2 \max_{1 \leqslant \tau \leqslant k} |\varepsilon(\tau)|] \tag{20}$$

From Lemma 3. 3. 6 in [5], the parameter estimate algorithm (12)~(13) has the following properties:

$$\lim_{k\to\infty} \| \theta(k) - \theta(k-1) \| = 0, \tag{21}$$

$$\lim_{k\to\infty} \frac{e^2(k)}{1 + c\Phi^{\mathrm{T}}(k)\Phi(k)} = 0, \quad c = \lambda_{\max}[M^{-1}(0)]. \tag{22}$$

Consider equation (19), (21)and (22), yields:

$$\lim_{k\to\infty} \frac{\varepsilon^2(k)}{1 + c\Phi^{\mathrm{T}}(k)\Phi(k)} = 0. \tag{23}$$

Note equation (20) and (23), according to Lemma 6. 2. 1 in [5] we know that $\{u(k)\}$ is a bounded sequence, since the process is stable, so $\{y(k)\}$ is also a bounded sequence; Substitution of equation (19) into (18) yields $y(\infty) = w(\infty)$ by setting $k \to \infty$.

# 3 Simulation

In this section, a computer simulated examples is used to illustrate the behaviour of the controller provided in the paper, the process considered here is:

$$y(k) + 0.36y(k-1) + 0.24y(k-2) = 0.5u(k-1) + 0.25u(k-2) + d(k) + \xi(k)$$

where

$$d(k) = \begin{cases} 8, & 20 < k < 100, \\ 0, & \text{otherwise}, \end{cases}$$

$$\xi(k) = \eta(k) + 0.7\eta(k-1)$$

and $\eta(k) < 0.5$ is a white noise, Assume the desired closed-loop polynomial $T(z^{-1}) = (1 - 0.5z^{-1})^2$, the initial input-output data is set to zero, the parameter esitmates are initialized to unit vector, the initial covariance matrix is taken as $10^5 I$, the weighting factor $\lambda = 1$ and $N = 3$. The input and output response are shown in Fig. 2 and Fig. 1.

From Fig. 1 we see that the algorithm drives the $y(k)$ to track $w(k)$ well and rejects the stochastic disturbance effectively. There is no static control error observed when the load disturbance occur.

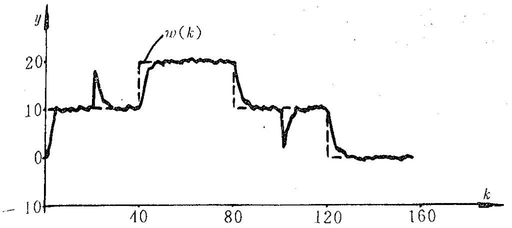


Fig. 1 Process cutnut $y(k)$ and reference $w(k)$

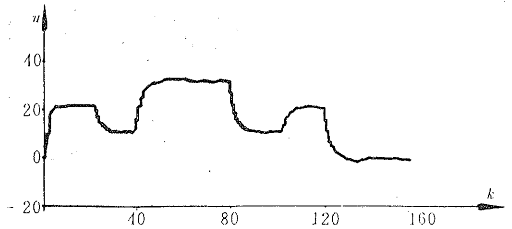


Fig. 2 Control signal

## 4 Conclusion

The self-tuning pole-placement controller based on long-range prediction of the process output has both the nice properties of long-range predictive controllers and the nice properties of pole-placement controllers, the closed-loop poles of the control system can be arbitrarily assigned and the computational load of the algorithm is not larger than the other predictive control algorithm. From the simulation studies it can be concluded that the controller seems to be fairly robust for load disturbance and stochastic process noise.

## References

[1] Clarke, D. W. et al. Generalized Predictive Control Part Ⅰ & Ⅱ, Automatica, 1987, 23(2):137—160

[2] De Keyser, R. M. C. et al.. A Comparative Study of Self-Adaptive Long-Range Predictive Control Mehtods. Automatica, 1988, 24(2):149—163

[3] Rouhani, R. et al.. Model Algorithmic Control: Basic Properties. Automatica, 1982, 18(4):401—414

[4] Lelic, M. A. et al.. Generalized Pole-Pleacement Self-Tuning Controller. Int. J. Control, 1987, 46(2):547—568

[5] Goodwin, G. C. and Sin, K. S.. Adaptive Filtering, Prediction and Control. Prentice-Hall, New Jersey, 1984

# 一种新的预测自校正控制器

毛志忠 杨 琳

(东北工学院自动控制系,110006)

**摘要**:本文通过在指标函数中引入加权多项式,为由脉冲响应模型描述的系统提出一种基于多步预测的极点配置自校正控制器,这种控制器能确保控制系统的稳定性,并能消除控制系统的稳态误差,算法的计算量也比其它的预测控制算法如 MAC,GPC 等小.

**关键词**:预测控制;极点配置;自校正控制;稳定性

## 本文作者简介

**毛志忠** 1961 年生.副教授.1984 年取得东北工学院工业自动化专业硕士学位,留校后主要从事计算机控制系统的研究和教学工作.1991 年获博士学位.现主要从事预测控制,自适应控制及过程控制系统的理论及应用研究.

**杨 琳** 1961 年生.工程师.1984 年获东北工学院学士学位,留校后主要从事计算机辅助设计实验教学工作.