

文章编号: 1001-0920(2005)03-0332-05

## 基于神经网络的非线性系统多步预测控制

张日东, 王树青

(浙江大学 先进控制技术研究所, 工业控制技术国家重点实验室, 浙江 杭州 310027)

**摘要:** 针对离散非线性系统, 利用非线性激励函数的局部线性表示, 提出一种可用于非线性过程的神经网络多步预测控制方法, 并给出了控制律的收敛性分析. 该方法将非线性系统处理成简单的线性和非线性两部分, 对复杂的非线性多步预测方程给出了直观而有效的线性形式, 并用线性预测控制方法求得控制律, 避免了复杂的非线性优化求解. 仿真结果表明了该算法的有效性.

**关键词:** 神经网络模型; 预测控制; 非线性过程

**中图分类号:** TP273      **文献标识码:** A

## Neural network based multi-step predictive control for nonlinear systems

ZHANG Ri-dong, WANG Shu-qing

(Institute of Advanced Process Control, Zhejiang University, Hangzhou 310027, China Correspondent: ZHANG Ri-dong, E-mail: zrd-el@163.com)

**Abstract** A neural network model-based multi-step predictive control algorithm with its convergence analysis for nonlinear discrete systems is presented. The structure of this nonlinear system is separated into a simple linear part and a nonlinear part by using the local linearization of nonlinear activation functions. The proposed algorithm gives a direct and effective multi-step predicting method. It uses simple linear predictive control methods to get the control law and avoids the complicated nonlinear optimization. Simulation results show the efficiency of this method.

**Key words:** neural network model; predictive control; nonlinear processes

### 1 引言

广义预测控制(GPC)<sup>[1]</sup>具有预测模型、滚动优化和反馈校正3个基本特征以及优良的控制性能和鲁棒性. 它被认为是具有代表性的预测控制算法之一, 已广泛应用于工业过程. 非线性系统很难得到精确的模型, 这对于研究非线性GPC是不利的. 近年来, 利用神经网络描述非线性系统并进行控制已成为研究的热点之一; 将神经网络应用于预测控制的研究也逐渐增多, 已出现了多种基于神经网络的非线性预测控制方法<sup>[2-8]</sup>. 这些方法的基本思想是首先利用神经网络对非线性系统建模, 然后利用非线性搜索或线性化等方法求取控制律. 为给出直观而有效的多步预测方程, 需要利用多个网络或复合网

络等来实现; 为克服非线性预测控制在一般情况下得不到最优解的难题, 在控制律的求解方面给出了多种方法. 这些方法为神经网络非线性预测控制的分析与综合提供了一个基本思路, 使得预测控制器的设计有了基本的依据.

上述算法各有其优点, 但也不同程度地存在结构复杂、运算量大的弱点. 研究表明, 在对非线性系统进行神经网络预测控制时, 还需解决以下问题: 1) 对多步预测给出直观而有效的方法; 2) 设法避免复杂的非线性求解; 3) 减少网络数目, 从而减少计算量.

本文试图仅用一个神经网络来综合非线性系统的多步预测控制, 这不仅是为降低系统结构的复杂

收稿日期: 2004-05-09; 修回日期: 2004-08-05

作者简介: 张日东(1977—), 男, 辽宁葫芦岛人, 博士生, 从事预测控制、自适应控制等研究; 王树青(1939—), 男, 浙江仙居人, 教授, 博士生导师, 从事模型化与优化控制、预测控制等研究.

性, 而且是为减轻运算负担 其主要思路是利用非线性激励函数的局部线性表示, 将非线性多步预测转化为一列简单直观的线性多步预测形式, 并用线性广义预测控制方法<sup>[9]</sup>求取控制律, 从而不需要复杂的非线性优化搜索等求解方法

## 2 非线性系统的模型及其表示

考虑如下被控系统

$$y(t) = f[y(t-1), \dots, y(t-n), u(t-d-1), \dots, u(t-d-m)] \quad (1)$$

其中:  $n$  和  $m$  分别为时间序列  $\{y(t)\}$  和  $\{u(t)\}$  的阶次,  $d+1$  为系统的时滞 该系统可用三层BP神经网络表示为

$$y(t) = g \left\{ \sum_{i=1}^I w_3(i) g \left[ \sum_{j=1}^m w_2(i, j) u(t-d-j) + \sum_{j=m+1}^{m+n} w_2(i, j) y(t+m-j) \right] \right\} \quad (2)$$

其中:  $w_2(i, j), w_3(i)$  ( $i = 1, \dots, I, j = 1, \dots, m+n$ ) 为层与层之间的连接权, 输入结点  $n+m$  个, 隐层结点  $I$  个, 输出结点 1 个;  $g$  为激励函数

$$g(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

下面对式(2)中的  $g$  进行处理, 令

$$s_3(t) = \sum_{i=1}^I w_3(i) g \left[ \sum_{j=1}^m w_2(i, j) u(t-d-j) + \sum_{j=m+1}^{m+n} w_2(i, j) y(t+m-j) \right], \quad (4)$$

$$s_{2i}(t) = \sum_{i=1}^m w_2(i, j) u(t-d-j) + \sum_{j=m+1}^{m+n} w_2(i, j) y(t+m-j), \quad (5)$$

则有  $y(t) = g[s_3(t)]$  利用泰勒展开, 其中心点设为  $s_{31}$ , 可得到

$$y(t) = g(s_{31}) + g'(s_{31}) [s_3(t) - s_{31}] + \epsilon_1(\theta(t)) = g(s_{31}) s_3(t) + g'(s_{31}) - g(s_{31}) s_{31} + \epsilon_1(\theta(t)) = g(s_{31}) \sum_{i=1}^I w_2(i) g[s_{2i}(t)] + g'(s_{31}) - g(s_{31}) s_{31} + \epsilon_1(\theta(t)). \quad (6)$$

其中:  $\theta(t)$  代表向量  $(y(t-1), \dots, y(t-n), u(t-d-1), \dots, u(t-d-m))$ ,  $\epsilon_1$  是函数符号. 定义  $\theta(t+i)$  为

$$y(t+i-1), \dots, y(t+i-n), u(t+i-d-1), \dots, u(t+i-d-m).$$

同理, 对  $g[s_{2i}(t)]$  也作类似处理 设中心点为  $s_{2i}$  ( $i = 1, \dots, I$ ), 则有

$$g[s_{2i}(t)] = g(s_{2i}) + g'(s_{2i}) [s_{2i}(t) - s_{2i}] + \epsilon_i(\theta(t)), \quad (7)$$

其中  $\epsilon_i$  ( $i = 1, \dots, I$ ) 是函数符号, 并令常数  $s_{31}$  和  $s_{2i}$  ( $i = 1, \dots, I$ ) 为 0

将式(7)代入式(6), 并将非线性部分  $\epsilon_1(\theta(t))$  和  $\epsilon_i(\theta(t))$  ( $i = 1, \dots, I$ ) 合并成  $\epsilon(\theta(t))$ , 则有

$$y(t) = \sum_{i=1}^I w_3(i) g(s_{31}) \{g(s_{2i}) + g'(s_{2i}) [s_{2i}(t) - s_{2i}]\} + g(s_{31}) - g(s_{31}) s_{31} + \epsilon(\theta(t)) = \sum_{i=1}^I w_3(i) g(s_{31}) g(s_{2i}) s_{2i}(t) + \left\{ \sum_{i=1}^I w_3(i) g(s_{31}) [g(s_{2i}) - g'(s_{2i}) s_{2i}] + g(s_{31}) - g(s_{31}) s_{31} \right\} + \epsilon(\theta(t)). \quad (8)$$

其中非线性部分

$$\epsilon(\theta(t)) = \sum_{i=1}^I w_3(i) g(s_{31}) \epsilon_i(\theta(t)) + \epsilon_1(\theta(t)). \quad (9)$$

为书写简便, 令式(8)中第2项为  $N$ , 第1项中  $w_3(i) g(s_{31}) g(s_{2i})$  为  $M_i$ , 则可得到式(8)的等价式

$$y(t) = \sum_{i=1}^I M_i s_{2i}(t) + N + \epsilon(\theta(t)) = \sum_{i=1}^I M_i \left[ \sum_{j=1}^m w_2(i, j) u(t-d-j) + \sum_{j=m+1}^{m+n} w_2(i, j) y(t-j+m) \right] + N + \epsilon(\theta(t)). \quad (10)$$

将  $y(t)$  写成离散差分方程形式, 有

$$y(t) = a_1 y(t-1) + \dots + a_n y(t-n) + b_0 u(t-d-1) + \dots + b_{m-1} u(t-d-m) + N + \epsilon(\theta(t)). \quad (11)$$

对比式(10)与式(11)的系数, 得

$$\begin{cases} a_1 = \sum_{i=1}^I M_i w_2(i, m+1), b_0 = \sum_{i=1}^I M_i w_2(i, 1), \\ a_2 = \sum_{i=1}^I M_i w_2(i, m+2), b_1 = \sum_{i=1}^I M_i w_2(i, 2), \\ \vdots \\ a_n = \sum_{i=1}^I M_i w_2(i, m+n), b_{m-1} = \sum_{i=1}^I M_i w_2(i, m). \end{cases} \quad (12)$$

至此, 已将非线性系统模型(2) 处理成式(11)的形式, 其线性部分系数由式(12) 求取, 非线性部分  $\epsilon(\theta(t))$  由式(9) 表示

当取  $g(x)$  的中心点  $s_{31}$  和  $s_{2i}(i = 1, \dots, I)$  为 0 时,  $g(0) = 0.25$ , 则式(10) 中  $M_i = 0.0625w_3(i)$ .

### 3 控制系统设计

采用自适应策略, 须在每次控制周期到来时, 首先辨识系统的模型参数, 即神经网络的权值  $w_2(i, j)$  和  $w_3(i)$  ( $i = 1, \dots, I, j = 1, \dots, m + n$ ); 然后基于这些参数组成系统模型(2). 利用第2节的方法进行处理, 辨识到的权值均为固定数值, 因此由  $N$  的定义知, 在每个控制周期  $N$  为某一常数, 但每个控制周期对应的  $N$  却是不同的数值. 若不采用自适应策略, 即认定模型(2) 是精确的, 则权值均为固定数值, 从而  $N$  为一个固定常数, 且在每个控制周期均不变. 无论是否采用自适应策略, 在每个控制周期用第2节方法处理模型时,  $N$  均为某一常数. 因此, 式(11) 可改写为

$$y(t) = A_1 y(t-1) + \dots + A_{n+1} y(t-n-1) + B_{1,0} \Delta u(t-d-1) + \dots + B_{1,m-1} \Delta u(t-d-m) + \Delta \epsilon(\theta(t)). \quad (13)$$

其中:  $A_1 = 1 + a_1, A_i = a_i - a_{i-1} (i = 2, \dots, n), A_{n+1} = -a_n, B_{1,i} = b_i (i = 0, \dots, m-1)$ .

将最优预测输出分成三部分: 第1部分取决于过去的输入和输出, 表示为  $Y_p$ ; 第2部分取决于现在和未来的输入, 表示为  $GU$ ; 第3部分为预测误差, 包括非线性误差  $E_1$  和外界干扰引起的误差  $E_2$ . 这里, 预测时域和控制时域均为  $p$ ,  $E_2$  是预测控制中的反馈校正项, 它包含  $p$  个  $y(t) - \hat{y}(t)$ , 因此得到

$$Y = Y_p + GU + E_1 + E_2 \quad (14)$$

其中

$$Y = [y(t+d+1/t), \dots, y(t+d+p/t)]^T, \\ Y_p = [y_p(t+d+1), \dots, y_p(t+d+p)]^T, \\ U = [\Delta u(t), \Delta u(t+1), \dots, \Delta u(t+p-1)]^T, \\ E_1 = [\epsilon(\theta(t+1)), \dots, \epsilon(\theta(t+p))]^T, \\ E_2 = [y(t) - \hat{y}(t), \dots, y(t) - \hat{y}(t)]^T,$$

$$G = \begin{bmatrix} B_{1,0} & & & \\ B_{2,0} & B_{1,0} & & 0 \\ \vdots & \vdots & \ddots & \\ B_{p,0} & B_{p-1,0} & \dots & B_{1,0} \end{bmatrix}.$$

式中:  $y(t)$  为实际输出;  $\hat{y}(t)$  为神经网络模型输出;  $Y_p$  为系统的自由输出, 可由其神经网络模型计算;  $G$  中的元素计算如下:

$$\begin{cases} B_{1,0} = b_0, \\ B_{k,0} = b_{k-1} + \sum_{j=1}^{k-1} A_j B_{k-j,0}, \\ k = 2, 3, \dots, p. \end{cases} \quad (15)$$

取参考轨迹

$$\begin{cases} y_r(t+d) = y(t), \\ y_r(t+d+k) = \alpha^k y(t) + (1-\alpha^k) y_s, \\ k = 1, 2, \dots, p-1. \end{cases} \quad (16)$$

其中:  $\alpha$  为输出柔化系数,  $y_s$  为设定值. 另外, 分别设参考轨迹向量和目标函数

$$\begin{cases} Y_r = [y_r(t+d+1), \dots, y_r(t+d+p)]^T, \\ J = \min \{ (Y_r - Y)^T (Y_r - Y) + \beta^2 U^T U \}. \end{cases} \quad (17)$$

其中  $\beta^2$  为加权项. 由  $\partial/\partial U = 0$  得

$$U = (G^T G + \beta^2 I)^{-1} G^T (Y_r - Y_p - E_1 - E_2). \quad (18)$$

设  $q^T$  为  $(G^T G + \beta^2 I)^{-1} G^T$  的第1行组成的行向量, 则  $t$  时刻的控制量为

$$u(t) = u(t-1) + q^T (Y_r - Y_p - E_1 - E_2). \quad (19)$$

由于系统为非线性系统,  $E_1$  是未知的, 本文采取迭代递推求取控制律, 算法为

$$\begin{cases} E_{10} = 0, \\ U_0 = (G^T G + \beta^2 I)^{-1} G^T (Y_r - Y_p - E_2 - E_{10}), \\ E_{1k+1} = E_{1k} + \delta(k) (Y_{mk} - Y_k), \\ U_{k+1} = (G^T G + \beta^2 I)^{-1} G^T (Y_r - Y_p - E_2 - E_{1k+1}). \end{cases} \quad (20)$$

其中:  $E_{10}$  为  $E_1$  的初始值;  $U_0$  为  $U$  的初始值; 下标  $k$  代表各分量第  $k$  步的值;  $Y_{mk}$  为  $U_k$  代入神经网络模型而求出的输出;  $\delta(k)$  为收敛因子, 它是正标量递减序列;  $\delta(k) \in (0, 1), \lim_{k \rightarrow \infty} \delta(k) = 0$ . 适当选取

$\delta(k)$ , 上述递推可以很快收敛, 通常取  $\delta(k) = 1/k$ . 当  $Y = Y_m$  时, 求出的  $U_k$  即为最优控制律. 实际上, 若  $E_{1k+1} - E_{1k}$  足够小, 则可认为  $U_k$  为控制律. 现将本文算法归纳如下:

- 1) 用输入输出样本对神经网络进行一段时间 的离线训练, 得到权值的初值;
- 2) 读取系统输出  $y(t)$ , 辨识网络的权值;
- 3) 利用第2节的方法重新表示系统;
- 4) 计算系统的自由输出  $Y_p$ ;
- 5) 用式(16) 计算参考轨迹;
- 6) 用式(20) 求取控制量并进行控制;
- 7) 返回到第2) 步.

### 4 收敛性分析

本节从分析  $E_{1k}$  的收敛性入手。由式(20)知, 对于充分大的  $k$ ,  $\delta(k)$  将任意小, 因此  $E_{1k}$  和  $U_k$  的变化都将越来越小。当  $E_{1k}$  收敛于真值  $\bar{E}_1$  时,  $Y_k$  将收敛于  $Y_{mk}$ , 从而得到最优控制律。若  $E_{1k}$  收敛于真值  $\bar{E}_1$ , 则由上述分析可知, 最优控制律的求解是收敛的。本文控制律的收敛性借助于伴随微分方程的稳定性来确定。

**引理 1**<sup>[10]</sup> 考虑如下递推算法:

$$\begin{cases} \alpha(k) = \alpha(k-1) + \delta(k)Q(k), \\ \alpha(k-1), \Phi(k), \\ \Phi(k) = A(\alpha(k-1))\Phi(k-1) + \\ B(\alpha(k-1))e(k). \end{cases} \quad (21)$$

其中  $\alpha(k)$  称为估计量。定义  $D_s = \{\alpha | A(\alpha)$  的所有特征值都严格在单位圆内 $\}$ ,  $D_R$  是  $D_s$  的一个连通开子集, 在子集  $D_R$  上, 式(21)中的函数满足正则条件  $C1 \sim C5$ <sup>[10]</sup>。对于式(21)算法的伴随微分方程  $\frac{d}{d\tau}\alpha_b(\tau) = f^*(\alpha_b(\tau))$ , 若存在一个正函数  $V(\alpha_b)$ , 使得

$$\begin{cases} \frac{d}{d\tau}V(\alpha_b) < 0, \forall \alpha_b \in D_A; \\ \frac{d}{d\tau}V(\alpha_b) = 0, \forall \alpha_b \in D_c, D_c \subset D_A. \end{cases} \quad (22)$$

则当  $k \rightarrow \infty$  时,  $\alpha(k)$  以概率 1 趋于不变集  $D_c$ , 即  $\alpha(k) \xrightarrow{w.p.1} D_c$ 。如果  $\alpha^*$  是微分方程的整体渐近稳定平衡点, 则当  $k \rightarrow \infty$  时,  $\alpha(k) \xrightarrow{w.p.1} \alpha^*$ 。

**定理 1** 对于非线性系统(1), 如果将其神经网络模型(2)表示为式(11)的形式, 控制器设计为式(20)的形式, 控制器设计参数  $\delta(k)$  是正标量递减序列, 且  $\delta(k) \in (0, 1), \lim_{k \rightarrow \infty} \delta(k) = 0$ , 则系统的预测控制律是收敛的。

**证明** 由式(20)和(14)得

$$\begin{aligned} E_{1k+1} &= E_{1k} + \delta(k) [Y_{mk} - Y_p - \\ &GU_k - E_{1k} - E_2] = \\ &FE_{1k} + D. \end{aligned} \quad (23)$$

其中  $D$  和  $F$  在每次递推时为某个常数矩阵, 即

$$\begin{aligned} D &= \delta(k) [Y_{mk} - Y_p - GU_k - E_2], \\ F &= \text{diag}(1 - \delta(k), \dots, 1 - \delta(k)). \end{aligned}$$

定义  $D_s = \{F | F$  的所有特征值都严格在单位圆内 $\}$ , 而

$$\lambda_k(F) = 1 - \delta(k) < 1, k = 1, \dots, p.$$

可见  $D_s$  为全平面, 即  $D_R = D_s = R^n$ 。由式(23)及  $\delta(k) = 1/k$ , 条件  $C1 \sim C5$  均可满足。于是, 式(23)的伴随微分方程为

$$\frac{dE_{1D}(\tau)}{d\tau} = f^*(E_{1D}(\tau)) =$$

$$\lim_k E\{Y_{mk} - Y_p - GU_k - E_{1D}(\tau) - E_2\},$$

而  $Y_{mk} = Y_p + GU_k + \bar{E}_1 + E_2$  进而有

$$f^*(E_{1D}(\tau)) = \bar{E}_1 - E_{1D}(\tau). \quad (24)$$

其中  $\bar{E}_1$  为  $E_1$  的真值。

设  $E_{1D}^*$  是微分方程(24)的平衡点, 则有  $E_{1D}^* = \bar{E}_1$ , 即平衡点位于  $E_1$  的真值上。

取微分方程(24)的 Lyapunov 函数

$$V(E_{1D}) = \frac{1}{2}(\bar{E}_1 - E_{1D})^T(\bar{E}_1 - E_{1D}) > 0,$$

则

$$\begin{aligned} \frac{d}{d\tau}V(E_{1D}) &= (\bar{E}_1 - E_{1D})^T \frac{d}{d\tau}(\bar{E}_1 - E_{1D}) = \\ &= -(\bar{E}_1 - E_{1D})^T(\bar{E}_1 - E_{1D}) < 0, \forall E_{1D}, \end{aligned}$$

且

$$\frac{d}{d\tau}V(E_{1D}) = 0, E_{1D} = E_{1D}^* = \bar{E}_1.$$

可见, 微分方程存在不变集  $D_c = \{\bar{E}_1\}$ , 而吸收域为全平面。由定理 1 知  $E_1$  收敛于  $\bar{E}_1$ 。

### 5 仿真研究

设被控对象的真实模型为

$$y(t) = \frac{0.91y(t-1) + u(t-5)}{1 + u(t-6)} + e(t)/\Delta$$

仿真是在 Matlab 6.5 环境下编程实现的, 其中给出了本文方法与一般 GPC 方法控制该对象的对比结果, 图 1 为一般 GPC 的仿真结果, 图 2 为本文方法的仿真结果。设定值取  $y_s = 0.5$ , 控制参数选  $p = 5, \beta^2 = 1, \alpha = 0.65, e(t)$  是  $-0.01 \sim 0.01$  均匀分布的独立的随机数。神经网络结构取 3-5-1, 而一般 GPC 采

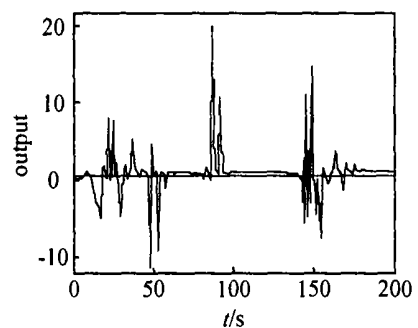


图 1 一般 GPC 的仿真结果

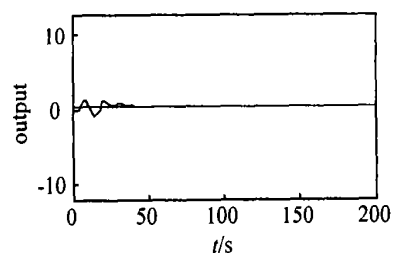


图 2 本文方法的仿真结果

## 用模型

$$y(t) = a_{1y}(t-1) + b_{1u}(t-5) + b_{2u}(t-6)$$

描述该对象,用递推最小二乘法来辨识模型的参数  $a_1, b_1, b_2$  可见,采用一般的 GPC 方法,控制过程中震荡剧烈,超调量大,而且最终会有很大的余差;而本文方法则不存在这些缺点,控制效果良好。

另外,本文只用一个网络并用线性方法给出了直观有效的多步预测及解析的控制律,比较圆满地解决了引言中提到的3个问题,与只解决这3个问题中的一个或两个的神经网络非线性预测控制算法相比,本文方法更具参考价值。

## 6 结 语

本文提出的神经网络非线性预测控制算法,将复杂的非线性多步预测转化为直观有效的线性形式,不需要非线性求解。由于仅用一个神经网络,不但降低了结构的复杂性,而且减轻了运算负担。仿真结果证明了该方法的有效性。

## 参考文献(References)

- [1] Clarke D W, Mohtadi C, Tuffs P S. Generalized predictive control — Part I: The basic algorithm [J]. *Automatica*, 1987, 23(2): 137-148
- [2] Schenker B, Agarwal M. Long-range predictive control for poorly known systems [J]. *Int J Control*, 1990, 62(1): 227-238
- [3] Zhang J, Morris A J. Long-range predictive models based on locally recurrent neural networks [A]. *Preprints of IFAC YAC 95* [C]. Beijing, 1995: 708-712
- [4] Sorensen P H, Norgaard M, Ravn O. Implementation of neural network based non-linear predictive control [J]. *Neurocomputing*, 1999, 28(1): 37-51
- [5] 张兴会,陈增强,袁著祉. 基于神经网络模型的非线性多步预测学习控制器 [J]. *控制与决策*, 2002, 17(增刊): 820-822  
(Zhang X H, Chen Z Q, Yuan Z Z. Nonlinear predictive controller based on neural networks [J]. *Control and Decision*, 2002, 17(Supp): 820-822)
- [6] 靳其兵,王建辉,顾树生. 多步预测性能指标函数下的神经网络逆动态控制方法 [J]. *控制与决策*, 1999, 14(4): 308-312  
(Jin Q B, Wang J H, Gu S S. Multi-step prediction cost function based reverse-dynamic method [J]. *Control and Decision*, 1999, 14(4): 308-312)
- [7] 杨煜普,黄新民,许晓鸣. 非线性系统多步预测控制的复合神经网络实现 [J]. *控制与决策*, 1999, 14(4): 314-318  
(Yang Y P, Huang X M, Xu X M. Nonlinear multi-step predictive control using compound neural networks [J]. *Control and Decision*, 1999, 14(4): 314-318)
- [8] 刘军,何星,许晓鸣. 基于神经网络非线性模型的多级工作点阶跃响应扩展DMC预测控制 [J]. *控制与决策*, 2000, 15(3): 342-344  
(Liu J, He X, Xu X M. Extension of DMC predictive control using neural network base nonlinear multi-step operation point responses [J]. *Control and Decision*, 2000, 15(3): 342-344)
- [9] 金元郁,顾兴源. 改进的广义预测控制算法 [J]. *信息与控制*, 1990, 19(3): 8-14  
(Jin Y Y, Gu X Y. Improved generalized predictive control [J]. *Information and Control*, 1990, 19(3): 8-14)
- [10] 方崇智,萧德云. *过程辨识* [M]. 北京: 清华大学出版社, 1998

(上接第331页)

- [2] Baik I C, Kim K H, Youn M J. Robust nonlinear speed control of PM synchronous motor using boundary layer integral sliding mode control technique [J]. *IEEE Transactions on Control System Technology*, 2000, 8(1): 47-54
- [3] Kaddouri A, Akhrif O, Lehuuy H. Adaptive nonlinear control for speed regulation of a permanent magnet synchronous motor [A]. *Proc of the 25th Annual Conf of the IEEE Industrial Electronics Society* [C]. California, 1999, 3: 1079-1084
- [4] Carroll J J, Dawson D M. Robust tracking control of a BLDC motor with application to direct-drive robotics [A]. *Proc of IEEE Int Conf on Robotics and Automation* [C]. Atlanta, 1993, 5: 94-99
- [5] 李三东,薛花,纪志成. 基于Matlab永磁同步电机控制系统的仿真建模 [J]. *江南大学学报*, 2004, 3(2): 115-120  
(Li S D, Xue H, Ji Z C. Modeling and simulation of PM SM control system based on Matlab [J]. *J of Southern Yangtze University*, 2004, 3(2): 115-120)