

文章编号: 1005-8893 (2006) 01-0045-04

对角递归神经网络在非线性内模控制中的应用

吴志敏

(江苏工业学院 计算机科学与工程系, 江苏 常州 213016)

摘要: 提出一种基于对角递归神经网络的非线性内模控制方案, 其内部模型和内模控制器的建立均由对角递归神经网络来实现, 同时改进神经网络控制器的训练函数。仿真结果表明, 改进的内模控制系统与基本 BP 神经网络内模控制系统相比, 具有跟踪速度快, 超调量小, 以及对对象参数摄动具有较好的适应性。

关键词: 对角递归神经网络; 内模控制; 非线性系统

中图分类号: TP 273 **文献标识码:** A

Investigation on Nonlinear Internal Model Control Based on Diagonal Recurrent Neural Network

WU Zhi-min

(Department of Computer Science and Technology, Jiangsu Polytechnic University, Changzhou 213016, China)

Abstract: Nonlinear internal model control scheme based on diagonal recurrent neural network is presented. Internal model and internal model controller are established by diagonal recurrent neural network. Meanwhile, this paper improves the training function of the neural network controller. The simulation results show that improved internal control has the faster tracking performance, less overshoot and better adaptability than the internal control based on BP neural network.

Key words: diagonal recurrent neural network; internal model control; nonlinear system

80年代初, Garcia 和 Morari 受模型算法控制和动态矩阵控制的启发, 提出了内模控制 (IMC) 理论^[1], 它具有设计简单、调节性能好、鲁棒性强并能消除不可测干扰的影响等优点。此后, Economou 等人又将内模控制扩展到非线性系统控制中^[2,3]。在非线性内模控制中, 实现控制的关键在于获取被控过程的模型及其逆模型。但对于非线性过程来说, 即使逆模型存在往往也很难直接获得。而神经网络因其具有良好的非线性函数逼近能力, 为非线性内模控制的研究开辟了一条新的途

径^[4,5]。

对角递归神经网络是一种特殊的动态神经网络, 是由 Chao-Chee Ku 等^[6~8]首次提出并对其收敛性进行了证明。它既具有一般动态网络易于处理动态非线性问题的特点, 又具有结构简单、容易构造训练算法等优点。因此, 本文将对角递归神经网络引入到非线性内模控制中, 建立非线性对象的内部模型及其内模控制器, 并将学习算法中待优化的目标函数进行了改进, 从而有效的实现了基于对角递归神经网络的内模控制。仿真结果表明, 该控

收稿日期: 2005-10-08

作者简介: 吴志敏 (1977-), 女, 辽宁锦州人, 硕士。

制方案具有良好的跟踪性和鲁棒性。

1 对角递归神经网络非线性内模控制

基于对角递归神经网络的非线性内模控制系统的结构如图 1 所示。图中 u 为控制信号, y_r 为参考输入, y 为被控对象的输出, y_m 为对角递归神经网络的输出, v 为系统的不可测干扰, $G_r(z^{-1})$ 为反馈滤波器, 可取为简单的一阶形式, 即 $G_r(z^{-1}) = \frac{1-\alpha_f}{1-\alpha_f z^{-1}}$, 适当选择滤波器的参数 α_f 可有效地抑制干扰, 增强系统的鲁棒性。这里被控对象的内部模型 (NNM) 和内模控制器 (NNC) 都用对角递归神经网络来实现, 其具体的实现过程如下。

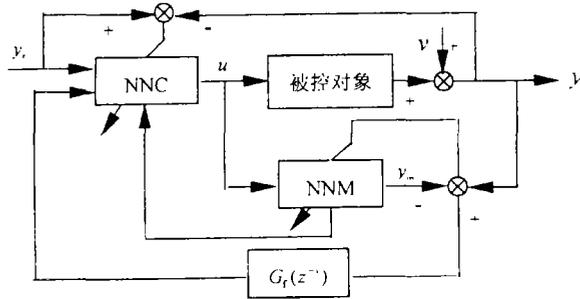


图 1 基于对角递归神经网络的内模控制结构

Fig.1 Structure of nonlinear internal model control based on diagonal recurrent neural netwo

1.1 神经网络内部模型 (NNM) 的建立

设被控对象可用如下的离散时间非线性系统^[9]表示:

$$y(k+1) = f[y(k), \dots, y(k-n+1), u(k), \dots, u(k-m+1)] + v(k) \quad (1)$$

其中 $u(k)$ 、 $y(k)$ 和 $v(k)$ 分别为系统的输入、输出和噪声, m 和 n 分别为输入和输出的阶次, $f(\cdot)$ 为非线性函数。这里采用单隐层的对角递归神经网络来建立被控对象的内部模型, 由于网络具有递归特性, 故其输入结构可简化为两节点输入, 即 $X^m = [X_1^m; X_2^m] = [y(k); u(k)]$ 。

隐含层:

$$S_j^m(k) = W_j^{dm} Z_j^m(k-1) + \sum_{i=1}^2 W_{ji}^{hm} X_i^m(k), \quad j=1, \dots, M \quad (2)$$

$$Z_j^m(k) = f(S_j^m(k)) = \frac{1}{1 + e^{-S_j^m(k)}}, \quad j=1, \dots, M \quad (3)$$

输出层:

$$y_m(k+1) = \sum_{j=1}^M W_j^{om} Z_j^m(k) \quad (4)$$

其中 $S_j^m(k)$ 、 $Z_j^m(k)$ 分别为隐层输入和输出; W_{ji}^{hm} , W_j^{dm} , W_j^{om} 分别表示输入层到隐层、递归层和隐层到输出层之间的连接权; M 为隐层节点数。训练 NNI 的性能指标函数定义为:

$$E_m(k+1) = \frac{1}{2} [y(k+1) - y_m(k+1)]^2 \quad (5)$$

利用动态 BP 算法修正连接权 W_{ji}^{hm} , W_j^{dm} , W_j^{om} 的公式如下:

$$W_j^{om}(k+1) = W_j^{om}(k) + \eta_m [y(k+1) - y_m(k+1)] Z_j^m(k) \quad (6)$$

$$W_j^{dm}(k+1) = W_j^{dm}(k) + \eta_m [y(k+1) - y_m(k+1)] W_j^{om}(k) P_j^m(k) \quad (7)$$

$$W_{ji}^{hm}(k+1) = W_{ji}^{hm}(k) + \eta_m [y(k+1) - y_m(k+1)] W_j^{om}(k) Q_{ji}^m(k) \quad (8)$$

$$P_j^m(k) = f'(S_j^m(k)) [W_j^{dm} P_j^m(k-1) + Z_j^m(k-1)] \quad (9)$$

$$Q_{ji}^m(k) = f'(S_j^m(k)) [W_j^{dm} Q_{ji}^m(k-1) + X_i^m(k)] \quad (10)$$

其中, η_m 为学习率, $f'(S_j^m(k)) = Z_j^m(k) (1 - Z_j^m(k))$, 初值选择为: $P_j^m(0) = 0$, $Q_{ji}^m(0) = 0$ 。

1.2 神经网络内模控制器 (NNC) 的建立

常规内模控制器由模型的逆构成, 物理实现困难, 本文采用对角递归神经网络来建立内模控制器。设被控对象的逆动态模型可用如下的离散时间非线性系统表示:

$$u(k) = f^{-1}[y(k+1), y(k), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1)] \quad (11)$$

在控制系统中总希望输出 y 能无偏差地跟踪参考输入 y_r , 因此可用 $y_r(k+1)$ 代替 $y(k+1)$, $y_r(k+1)$ 为下一时刻系统的参考输入, 可通过对当前及以前时刻参考输入值的线性化预测可以得到 $y_r(k+1)$ 的预测值。另外, 实际过程中, 由于过程输出的测量带有一定的误差, 故在建立逆模型的过程中, 用过程模型输出 y_m 代替过程输出 y , 故式 (11) 又可表示为:

$$u(k) = f^{-1}[y_r(k+1), y_m(k), \dots, y_m(k-n+1), u(k-1), \dots, u(k-m+1)] \quad (12)$$

为使内模控制系统更有效地抑制干扰、增强鲁

棒性, 在反馈回路中需引入滤波器, 因此, 还需加入一经过反馈滤波器的反馈量 $e_m \cdot G_f =$

$$(y(k) - y_m(k)) \cdot \left(\frac{1 - \alpha_f}{1 - \alpha_f z^{-1}} \right), \text{ 同时考虑到对}$$

角递归神经网络的递归特性, 故其输入层节点为:
 $X^c = [X_1^c; X_2^c; X_3^c; X_4^c; X_5^c] = [y_r(k+1); y_m(k); y_m(k-1); u(k-1); \alpha_f X_5^c(k-1) + (1 - \alpha_f)(y(k) - y_m(k))]$ (13)

隐含层:

$$S_j^c(k) = W_j^{dc} Z_j^c(k-1) + \sum_{i=1}^5 W_{ji}^{hc} X_i^c(k), \quad j=1, \dots, N \quad (14)$$

$$Z_j^c(k) = f(S_j^c(k)) = \frac{1}{1 + e^{-S_j^c(k)}}, \quad j=1, \dots, N \quad (15)$$

输出层:

$$u(k) = \sum_{j=1}^N W_j^{oc} Z_j^c(k) \quad (16)$$

其中, $S_j^c(k)$ 、 $Z_j^c(k)$ 分别为隐层输入和输出; W_{ji}^{hc} 、 W_j^{dc} 、 W_j^{oc} 分别表示输入层到隐层、递归层和隐层到输出层之间的连接权; N 为隐层节点数。为了改善控制系统的性能, 将神经网络控制器 (NNC) 的训练函数选为误差增量和输入增量的加权和: 即

$$E_c(k+1) = \frac{1}{2} [p(y_r(k+1) - y(k+1))^2 + q(u(k) - u(k-1))^2] \quad (17)$$

其中, p 和 q 分别为误差增量和输入增量的加权系数。

利用动态 BP 算法可得连接权 W_{ji}^{hc} 、 W_j^{dc} 、 W_j^{oc} 的学习算法如下:

$$\begin{aligned} \frac{\partial E_c}{\partial W_j^{oc}} &= -p(y_r(k+1) - y(k+1)) - \\ & y(k+1) \frac{\partial y(k+1)}{\partial u(k)} \frac{\partial u(k)}{\partial W_j^{oc}} + q(u(k) - \\ & u(k-1)) \frac{\partial u(k)}{\partial W_j^{oc}} = \\ & -[p(y_r(k+1) - y(k+1)) \frac{\partial y(k+1)}{\partial u(k)} - q(u(k) - \\ & u(k-1))] \frac{\partial u(k)}{\partial W_j^{oc}} = -p[(y_r(k+1) - \\ & y(k+1)) \frac{\partial y(k+1)}{\partial u(k)} - q(u(k) - u(k-1))] Z_j^c(k) \end{aligned} \quad (18)$$

$$W_j^{oc}(k+1) = W_j^{oc}(k) - \eta_c \frac{\partial E_c}{\partial W_j^{oc}} \quad (19)$$

同理可得:

$$\begin{aligned} \frac{\partial E_c}{\partial W_j^{dc}} &= -p[(y_r(k+1) - y(k+1)) \frac{\partial y(k+1)}{\partial u(k)} - \\ & q(u(k) - u(k-1))] W_j^{oc}(k) P_j^c(k) \end{aligned} \quad (20)$$

$$W_j^{dc}(k+1) = W_j^{dc}(k) - \eta_c \frac{\partial E_c}{\partial W_j^{dc}} \quad (21)$$

$$\begin{aligned} \frac{\partial E_c}{\partial W_{ji}^{hc}} &= -[p(y_r(k+1) - \\ & y(k+1)) \frac{\partial y(k+1)}{\partial u(k)} - q(u(k) - \\ & u(k-1))] W_j^{oc}(k) Q_{ji}^c(k) \end{aligned} \quad (22)$$

$$W_{ji}^{hc}(k+1) = W_{ji}^{hc}(k) - \eta_c \frac{\partial E_c}{\partial W_{ji}^{hc}} \quad (23)$$

其中 $P_j^c(k) = f'(S_j^c(k)) [W_j^{oc} P_j^c(k-1) + Z_j^c(k-1)]$, $Q_{ji}^c(k) = f'(S_j^c(k)) [W_j^{dc} Q_{ji}^c(k-1) + X_i^c(k)]$ 。

上述公式中, 包含了被控对象的偏导数, 但对于未知对象, 它是未知的。由于神经网络经过有限次学习后, $y_m(k+1)$ 能准确地逼近 $y(k+1)$, 因此可以用 $\frac{\partial y_m(k+1)}{\partial u(k)}$ 近似代替 $\frac{\partial y(k+1)}{\partial u(k)}$, 则有:

$$\begin{aligned} \frac{\partial y(k+1)}{\partial u(k)} &= \frac{\partial y_m(k+1)}{\partial u(k)} = \sum_{j=1}^M \frac{\partial y_m(k+1)}{\partial Z_j^m(k)} \\ \frac{\partial Z_j^m(k)}{\partial S_j^m(k)} \frac{\partial S_j^m(k)}{\partial u(k)} &= \sum_{j=1}^M W_j^{om} f'(S_j^m(k)) W_{jm}^{km} \end{aligned} \quad (24)$$

将式 (24) 分别代入式 (18)、(20)、(22) 即可得到对角递归神经网络内模控制器的权值修正公式。

2 仿真实例

为了验证本文提出的基于对角递归神经网络的非线性内模控制方案的有效性, 选择如下非线性系统进行仿真研究。

$$\begin{aligned} y(k+1) &= \frac{(1 + \delta_1) y(k) y(k-1)}{1 + y(k)^2 + y(k-1)^2} + \\ & (1 + \delta_2) u(k) + 0.5u(k-1) \end{aligned} \quad (25)$$

其中 δ_1 、 δ_2 表示参数变化量, 初始 $\delta_1 = \delta_2 = 0$ 。

图 2 所示为给定期望输出信号为单位阶跃信号, 在 $k=30$ 时对象参数发生变化, 即 $\delta_1 = 0.6$, $\delta_2 = -0.35$, 并且在 $k=60$ 时, 加入大小为 0.1 的定植扰动, 此时本文提出的改进的对角递归神经网络内模控制与基本 BP 神经网络内模控制的阶跃响应曲线。其中对角递归神经网络内部模型和内模控制器的参数分别为 $\eta_m = 0.6$, $\eta_c = 0.1$, $p = 0.7$,

$q=0.6, \alpha_t=0.8$; 基本 BP 神经网络内模控制参数分别为 $\eta_m=0.6, \eta_c=0.14, \alpha_t=0.8$ 。

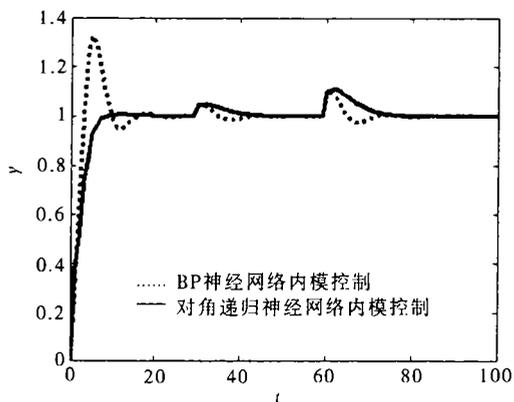


图2 系统阶跃响应曲线

Fig. 2 The curve of system step respond

图3所示为参考输入是周期为100步,幅值是 $[0.2, 0.6]$ 的方波,且在 $k=120$ 时对象参数发生变化,即 $\delta_1=0.6, \delta_2=-1$ 时的两种控制方案的输出曲线。其中对角递归神经网络内部模型和内模控制器的参数分别为 $\eta_m=0.09, \eta_c=0.01, p=0.7, q=0.6, \alpha_t=0.6$;基本BP神经网络内模控制参数分别为 $\eta_m=0.3, \eta_c=0.4, \alpha_t=0.6$ 。

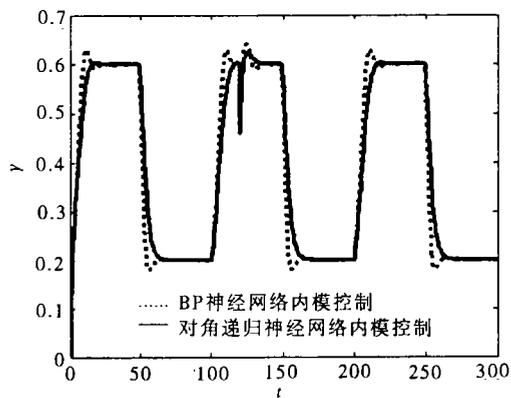


图3 系统的方波跟踪曲线

Fig. 3 System square-wave track curve

3 结论

本文采用对角递归神经网络建立被控对象的内部模型和内模控制器,并改进神经网络内模控制器的训练函数,进而实现了基于对角递归神经网络非线性内模控制。仿真结果表明,该控制系统不仅具有跟踪速度快,超调量小的优点,而且对参数扰动及突加干扰均具有良好的适应能力。

参考文献:

[1] Garcia C E, Morari M. Internal Model Control 1: An Unifying Review and Some New Results [J]. Ind Eng Chem Process Dev, 1982, 21 (2): 308-323.

[2] Economou C G, Morari M. Internal Model Control .5. Extension to Nonlinear Systems [J]. Ind Eng Chem Process Dev, 1986, 25: 403-411.

[3] Matausek M R, Stipanovic D M. Modified Nonlinear Internal Model Control [J]. Control and Computers, 1998, 26 (2): 57-63.

[4] Matausek M R, Miljkovic D M. Nonlinear Multi-Input-Multi-Output Neural Network Control of DC Motor Drive with Field Weakening [J]. IEEE Transactions on Industrial Electronics, 1998, 45: 185-187.

[5] Kambhampati C, Tham M, Warwick K, et al. Inverse Model Control Using Recurrent Networks [J]. Mathematics and Computers in Simulation, 2000, 51 (3-4): 181-199.

[6] Ku C C, Lee K Y. Diagonal Recurrent Neural Networks for Dynamic System Control [J]. IEEE Trans on Neural Network, 1993, 6 (1): 144-156.

[7] 张兴, 季建强, 张崇巍, 等. 基于内模控制的三相电压型 PWM 整流器不平衡控制策略研究 [J]. 中国电机工程学报, 2005, 25 (13): 51-57.

[8] 刘红军, 赵鑫, 薛浩洋. 采用 Adaline 神经网络实现增益自适应内模控制 [J]. 系统仿真学报, 2005, 17 (8): 1 955-1 959.

[9] Krishnapura V G, Jutan A. A Neural Adaptive Controller [J]. Chemical-Engineering Science, 2000, 55 (18): 3 803-3 812.