

一种自学习模糊神经网络多变量自适应控制器

王笑颜 符雪桐

(浙江大学工业控制技术国家重点实验室·杭州, 310027)

摘要: 本文在将文[6]的参数学习算法推广到多变量系统和对爬山法加以改进的基础上,提出了一种新的基于 Pi-sigma 混合型自适应模糊神经网络的多变量自适应控制器.该控制器能在不需过多先验知识的情况下在线自学习前件和后件参数.仿真结果表明,该控制器是有效的.

关键词: 自适应控制; 多变量系统; Pi-sigma 混合型神经网络; 自学习

A Self-Learning Multivariable Adaptive Controller Based on Fuzzy Logic Neural Network

Wang Xiaoyan and Fu Xuetong

(National laboratory of Industrial Control Technology, Zhejiang University·Hangzhou, 310027, P. R. China)

Abstract: In this paper, the learning algorithm in paper [6] is extended to multivariable system and the hill-climbing search algorithm is improved. Furthermore, a novel multivariable adaptive controller based on hybrid Pi-sigma neural network is proposed, which can learn the parameters of the IF and THEN part of the rules on line with little prior knowledge. The adaptive controller performs encouraging results in the simulation.

Key words: adaptive control; multivariable system; hybrid pi-sigma neural network; self-learning

1 引言(Introduction)

针对大量复杂程度高、参数突变、环境未知或对象数学模型难以得到的生产过程,人们提出了一些自适应控制器^[1-3].但是,当多变量系统存在耦合作用而无法分割成几个互相独立的单变量系统时,不能简单地套用单变量自适应控制器的设计方法.这时,就必须采用多变量控制策略^[4],设计相应的多变量自适应控制器.

另一方面,由于人工神经网络具有自学习的能力,将人工神经网络与模糊理论结合起来,可以克服传统的模糊控制方法难以对模型参数进行有效辨识^[6]的缺点.目前,模糊控制与神经网络的结合已成为智能控制的热点^[5].

为此,本文将文[6]中 Pi-sigma 混合型自适应模糊神经网络推广应用于多变量系统的控制,提出了一种新的模糊神经网络多变量自适应控制器(下文简称 MAFNC).

2 多变量自适应模糊神经网络控制器的结构 (Structure of the multivariable adaptive controller based on fuzzy logic neural network (MAFNC))

1985 年,Tagaki 和 Sugeno 提出了一种模糊推理模型^[7].为使其适用于多变量系统,现将其推广到多变量输出形式:

$$\begin{aligned} R^i: & \text{IF } x_1 \text{ is } A_1^i \text{ and } \dots \text{ and } x_k \text{ is } A_k^i \\ & \text{THEN } y_1^i = g_1^i(x_1, x_2, \dots, x_k) \\ & \text{and } \dots \text{ and } y_n^i = g_n^i(x_1, x_2, \dots, x_k) \end{aligned} \quad (1)$$

其中

$R^i (i = 1, 2, \dots, m)$ 为第 i 条规则;
 m 为模糊推理模型的规则总数;
 $x_I (I = 1, 2, \dots, k)$ 为前提变量;
 $y_j^i (j = 1, 2, \dots, n)$ 为第 i 条规则的结论;
 $A_I^i (I = 1, 2, \dots, k)$ 为模糊子集.

设输入为 $(x_1^0, x_2^0, \dots, x_k^0)$, 则输出为:

$$u_j = \frac{\sum W_i y_j^i}{\sum W_i}, (i = 1, 2, \dots, m, j = 1, 2, \dots, n) \quad (2)$$

$$W_i = \Lambda_{\mu_{A_I^i}}(x_I^0), (i = 1, 2, \dots, m, I = 1, 2, \dots, k) \quad (3)$$

其中, $\mu_{A_I^i}(x_I^0)$ 为连续分段多项式函数型的隶属函数^[6].

本文以 2 输入、2 输出多变量对象为例来设计相应的自适应控制器. 设 e_1, e_2 是设定值 y_1^d, y_2^d 与系统实际输出 y_1, y_2 的偏差, ce_1, ce_2 为偏差变化率.

基于 Pi-sigma 混合型模糊神经网络的 MAFNC

如图 1 所示. 该网络不仅包含加节点“ Σ ”和乘节点“ Π ”, 还包含模糊节点“ Λ ”. 模糊节点能完成基本的模糊操作, 如取小、取大等(如图 1), 假设后件子网有 h 个隐含节点, 那么整个神经网络的输出为:

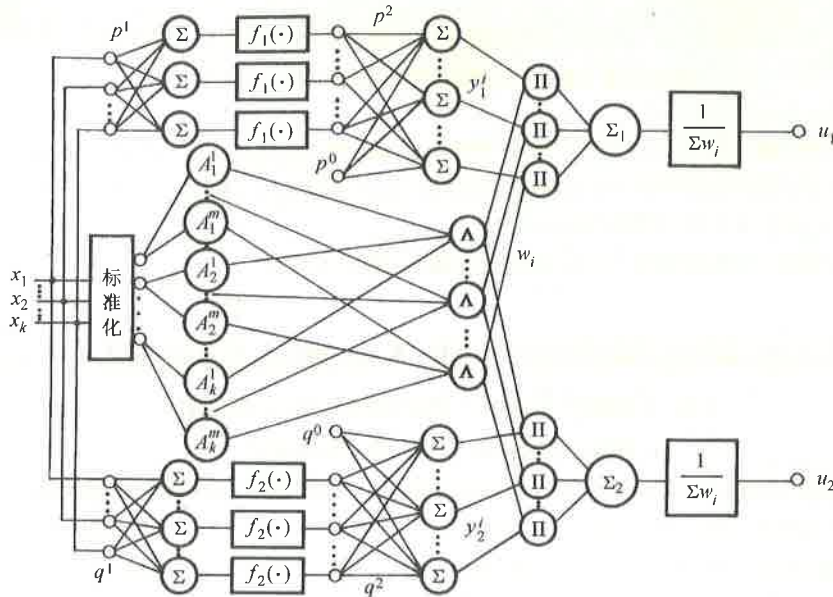


图 1 控制器结构

Fig. 1 Structure of the controller

$$u_j = \sum W_{ij} y_j^i / \sum W_i, (i = 1, 2, \dots, m, j = 1, 2) \quad (4)$$

$$y_1^i = g_1^i(x_1, x_2, \dots, x_k) = p_i^0 + \sum_{j=1}^h p_{ij}^2 f_1(\sum_{l=1}^k p_{lj}^1 x_l), (i = 1, 2, \dots, m) \quad (5a)$$

$$y_2^i = g_2^i(x_1, x_2, \dots, x_k) = q_i^0 + \sum_{j=1}^h q_{ij}^2 f_2(\sum_{l=1}^k q_{lj}^1 x_l), (i = 1, 2, \dots, m) \quad (5b)$$

$$W_i = \Delta \mu_{A_i^i}(\bar{x}_I), (i = 1, 2, \dots, m, I = 1, 2, \dots, k) \quad (6)$$

式中 u_1, u_2 为控制器输出, $f_1(\cdot), f_2(\cdot)$ 为 S 型非线性函数, \bar{x}_I 为 x_I 的标准化值(如图 1). 对比式(4) ~ (6) 与式(1) ~ (3) 可知, 当式(1) ~ (3) 所表示的系统是二维时, 两者是等价的. 为了使神经网络的学习更有效率, 隶属函数可取为 Gaussian 型^[6]:

$$\mu_{A_i^i}(\bar{x}_I) = \exp(-\frac{(\bar{x}_I - a_i^i)^2}{b_i^i}), b_i^i > 0. \quad (7)$$

式中 a_i^i, b_i^i 是描述隶属函数形状的两个参数, 其变化将影响网络输出, 从而影响控制效果. 下文将提出一种学习算法, 在设定两个初值后, 即可通过自动在线学习, 将它们调整至最优值.

3 多变量自适应模糊神经网络控制器的学习算法(Learning algorithm of the MAFNC)

1) 后件参数学习算法.

通常取误差目标函数为:

$$E = \frac{1}{2} [(u_1^d - u_1)^2 + (u_2^d - u_2)^2], \quad (8)$$

式中, u_1^d, u_2^d 为控制器的期望输出; u_1, u_2 为控制器实际输出. 由于在实际控制系统中难以得到 u_1^d, u_2^d , 本文取控制系统输出误差为误差目标函数:

$$E_p = \frac{1}{2} [(y_1^d - y_1)^2 + (y_2^d - y_2)^2], \quad (9)$$

其中 y_1^d, y_2^d 为系统输出设定值, y_1, y_2 为系统实际输出, 是 u_1, u_2 的函数.

由于乘节点的存在, 误差不能反传, 但假如把 W_i 看作为后件节点“ Σ ”和终点节点“ Σ_j ”($j = 1, 2$) 之间的权, 则乘节点可忽略^[6]. 这样由 BP 算法可得 p_i^0 的学习算法:

$$\begin{aligned} \Delta p_i^0 &= \eta \frac{\partial E_p}{\partial p_i^0} = -\eta \frac{\partial E_p}{\partial u_1} \frac{\partial u_1}{\partial p_i^0} = \\ &= -\eta \left[\frac{\partial E_p}{\partial y_1} \frac{\partial y_1}{\partial u_1} + \frac{\partial E_p}{\partial y_2} \frac{\partial y_2}{\partial u_1} \right] \frac{\partial u_1}{\partial y_1^i} \frac{\partial y_1^i}{\partial p_i^0} = \\ &= \eta \left[(y_1^d - y_1) \frac{\partial y_1}{\partial u_1} + (y_2^d - y_2) \frac{\partial y_2}{\partial u_1} \right]. \end{aligned}$$

$$\frac{W_i}{\sum W_i} \text{ 记为 } \eta \Delta_i^i, \quad (10)$$

式中 η 学习率, $\partial y_i / \partial u_1 \approx \Delta y_i / \Delta u_1 (i = 1, 2)$, 可由计算机辅助计算得到. 同样可得

$$\Delta p_{ij}^1 = -\eta \frac{\partial E_p}{\partial p_{ij}^1} = -\eta \frac{\partial E_p}{\partial u_1} \frac{\partial u_1}{\partial p_{ij}^1} = \eta \sum_{i=1}^m \Delta_i^i p_{ji}^2 x_i f_1', \quad (11)$$

$$\Delta p_{ij}^{2'} = -\eta \frac{\partial E_p}{\partial p_{ij}^{2'}} = \eta \Delta_i^i f_1. \quad (12)$$

同理易得 q_i^0, q_{ij}^1 和 q_{ij}^2 的学习算法.

2) 隶属函数参数学习算法.

为了更清楚地说明问题, 现将式(1)中的输入简化为两个变量 x_1, x_2 , 把它们分为 N_1, N_2 个模糊子集, 并设模糊推理模型共有 M 条 ($M = N_1 \times N_2$) 规则. 由式(6)可得:

$$W_i = \mu_{A_1^i}(\bar{x}_1) \Delta \mu_{A_2^i}(\bar{x}_2) \text{ 记为 } C_i \Delta D_i, \quad (13)$$

修正 C_i 或 D_i 的参数都可能改变 W_i , 下文讨论选择哪一项进行参数修正及如何修正的问题.

利用文[6]的爬山法可以解决这一问题, 但因其计算量很大, 而且因无法预知网络的输出期望值而无法计算网络输出误差, 另外, 生产过程也不允许测量过多的误差数据, 所以, 直接采用此法不是很适用. 故对其作如下改进:

当把 W_i 看作连接权时, 根据 BP 算法可得到为使误差 E_p 下降 W_i 所需修正的值:

$$\Delta W_i = -\eta \frac{\partial E_p}{\partial W_i} = \eta \left\{ (y_1^d - y_1) \frac{\partial y_1}{\partial u_1} + (y_2^d - y_2) \frac{\partial y_2}{\partial u_1} \sum \frac{y_1^i}{W_i} + \left[(y_1^d - y_1) \frac{\partial y_1}{\partial u_2} + (y_2^d - y_2) \frac{\partial y_2}{\partial u_2} \right] \sum \frac{y_2^i}{W_i} \right\} \text{ 记为 } \eta \Delta_b^i. \quad (14)$$

由式(13), 首先试算修正 C_i 项的参数而 D_i 项的参数保持不变时 W_i 的变化量 ΔW_{i1} ; 再试算修正 D_i 项的参数而 C_i 项的参数保持不变时 W_i 的变化量 ΔW_{i2} . 比较 ΔW_{i1} 与 ΔW_{i2} , 若 W_{i1} 与 W_i 较接近, 则修正 C_i 项的参数; 反之, 修正 D_i 项的参数. 这里所采用的修正算法根据 BP 算法经公式推导为(设 β 为学习率):

$$\Delta a_j^i = \beta \Delta_B \frac{2(\bar{x}_j - a_j^i)}{b_j^i} \exp \left[-\frac{(\bar{x}_j - a_j^i)^2}{b_j^i} \right], \quad (15)$$

$$\Delta b_j^i = \beta \Delta_B \left[\frac{(\bar{x}_j - a_j^i)}{b_j^i} \right]^2 \exp \left[-\frac{(\bar{x}_j - a_j^i)^2}{b_j^i} \right]. \quad (16)$$

假设被选定修正隶属函数参数的模糊子集为 $A_j^i (i = 1, \dots, M; j = 1, 2)$, 当 a_j^i, b_j^i 修正后, 所有规则中的模糊子集 A_j^i 的隶属函数参数都要更新为 a_j^i, b_j^i 修正后的值(如图 1 所示), 再进行下一次学习.

4 仿真结果(Result of simulation)

取仿真对象为大修前锅炉数学模型

$$G(s) = \begin{bmatrix} \frac{0.5}{2s^2 + 2s + 1} & \frac{-0.1}{5s^2 + 3s + 1} \\ \frac{-1.1}{7s + 1} & \frac{0.6}{8s + 1} \end{bmatrix}.$$

为克服 BP 算法收敛速度慢的缺点, 在仿真中采用文[8]的 PID 型 BP 算法(其收敛速度较一般 BP 算法快 22 倍), 得到下列三种情况的控制系统响应曲线(如图 2 所示):

- 1) $\Delta y_1^d = 1, \Delta y_2^d = 0,$
- 2) $\Delta y_1^d = 0, \Delta y_2^d = 1,$
- 3) $\Delta y_1^d = 1, \Delta y_2^d = 1.$

图 2 中曲线显示有控制静差, 在实际应用中, 可通过提高设定的控制精度来减小, 以满足预定的要求. 综上所述, 本文提出的自学习模糊神经网络多变量自适应控制器是有效的.

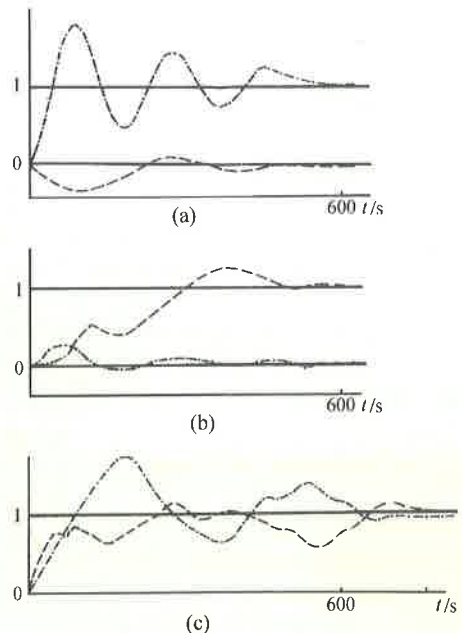


图 2 控制系统响应曲线

Fig. 2 Responding track of the controlled system

5 结论(Conclusion)

本文将文[6]的参数学习算法推广到多变量系统, 同时对爬山法进行了改进, 并在此基础上提出了

一种新的多变量自适应控制器. 该控制器能在不需过多先验知识的情况下在线自学习前件和后件参数. 仿真结果表明, 该控制器是有效的.

参考文献(References)

- 1 Chen Chun Lee. A Self-learning rule-based controller employing approximate reasoning and neural net concepts. *Int. J. Intelligent System*. 1991, 16(1): 71 - 93
- 2 Psaltis D, Sideris A and Yamamura A A. A multilayered neural network controller. *IEEE Control System Mag.*, 1987, 8(4): 17 - 21
- 3 Chen V C and Pao Y H. Learning control with neural networks. *Proc. 1989 IEEE Conf. on Robotics and Automation, Scottsdale, Arizona, U. S. A.*, 1989, 1448 - 1453
- 4 张化光等. 多变量模糊控制的现状与发展(I), *控制与决策*, 1995, 10(3): 193 - 203
- 5 Austrom K J. Directions in Intelligent Control. In: *Plenary Session IFAC International Symposium. ITAC'91, Tokyo, 1991*, 15 - 17
- 6 Jin Yaochu, Jiang Jingping and Zhu Jing. Adaptive fuzzy modelling and identification with its applications. *Int. J. System SCI*, 1995, 26(2): 197 - 212
- 7 Tagaki T and Sugeno M. Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. on Systems, Man, Cybern.*, 1985, SMC-15(1): 116 - 132
- 8 邓志东, 孙增圻, 刘建伟. BP网络的PID型二阶快速学习算法. *自动化学报*, 1995, 21(1): 67 - 71

本文作者简介

王笑颜 1972年生. 浙江大学工业控制技术研究所硕士研究生. 主要研究方向为智能控制.

符雪桐 1937年生. 浙江大学工业控制技术研究所教授级高工. 主要研究方向为大系统理论及智能控制.