

基于 BP 神经网络的自适应控制

谭永红

(桂林电子工业学院计算机系·桂林, 541004)

摘要: 本文利用 BP 神经网络对被控对象进行在线辨识和控制. 为实现自适应控制, 本文对 specialised learning 算法进行了改进. 在此基础上, 本文还提出了一种基于 BP 网络的自适应 PID 控制器.

关键词: 神经网络; 自适应控制; 学习

1 引言

近年来, 采用 BP 神经网络的控制方法已日益引起人们的重视. 由于 BP 网络可以表示任意非线性函数^[2], 并具有自适应学习、并行分布处理和有较强的鲁棒性及容错性等特点, 因此适用于对复杂非线性系统进行建模和控制.

目前, 用于训练 BP 神经网络控制器的方法主要有 general learning 和 specialised learning 两种^[1,3]. 前者需离线学习, 得到的是静态控制器^[3], 故不能适应系统出现的新情况及新变化, 而后者将被控对象作为 BP 网络的输出层, 可通过边学习边控制的方式对系统进行动态控制. 但是, 文献[1]提出的 specialised learning 方法要用到变量 $\frac{\partial y}{\partial u}$ (y 为对象输出, u 为控制量). 当对象特性未知时, $\frac{\partial y}{\partial u}$ 难于求得. 文献[3]用 $\text{sign}\left(\frac{\partial y}{\partial u}\right)$ 来代替 $\frac{\partial y}{\partial u}$ 但 $\text{sign}\left(\frac{\partial y}{\partial u}\right)$ 显得过于粗糙, 故其学习精度较差. 本文用神经网络模型输出 \hat{y} 关于控制 u 的灵敏度 $\frac{\partial \hat{y}}{\partial u}$ 来近似 $\frac{\partial y}{\partial u}$. 由此得到基于 BP 网络的自适应控制. 在此基础上, 还演变得到一种基于 BP 网络的自适应 PID 控制器.

2 基于 BP 神经网络的自适应控制

设被控对象是 SISO 离散时间非线性系统, 即

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

其中 n, m 分别为时间序列 $\{y(t)\}$ 和 $\{u(t)\}$ 的阶次. 设用来在线辨识对象特性的神经网络模型是个三层 BP 网络, 即

$$\text{输入层: } o_{1i}(t) = \begin{cases} y(t-i+1), & 1 \leq i \leq n, \\ u(t-i+n+1), & n+1 \leq i \leq m_1. \end{cases} \quad (2)$$

其中 $m_1 = n + m$.

$$\text{隐层: } nt_{2i}(t) = \sum_{j=1}^{m_1} w_{2ij} o_{1j}(t) + \theta_{2i}; \quad i = 1, 2, \dots, m_2, \quad (3)$$

$$o_{2i}(t) = g[nt_{2i}(t)]; \quad i = 1, 2, \dots, m_2, \quad (4)$$

输出层:
$$nt_3(t) = \sum_{j=1}^{m_2} w_{3j} o_{2j}(t) + \theta_3, \quad (5)$$

$$\hat{y}(t+1) = g[nt_3(t)]. \quad (6)$$

其中 w_{2ij}, w_{3j} 为层与层之间的两个神经元间的连接权, θ_{2i} 和 θ_3 为阈值、激励函数

$$g(x) = \frac{2}{1 + e^{-x}} - 1, \quad (7)$$

用 BP 学习算法来修正权及阈值, 使指标函数

$$J_1 = \frac{1}{2} [y(t) - \hat{y}(t)]^2 \quad (8)$$

最小化, 相应的修正公式为:

$$\delta(t) = [y(t) - \hat{y}(t)] g'[nt_3(t)], \quad (9)$$

$$W_{3j}(t+1) = W_{3j}(t) + \eta \delta(t) o_{2j}(t); \quad j = 1, 2, \dots, m_2, \quad (10)$$

$$\theta_3(t+1) = \theta_3(t) + \eta \delta(t), \quad (11)$$

$$W_{2ij}(t+1) = W_{2ij}(t) + \eta g'[nt_{2i}(t)] \delta(t) W_{3i}(t) o_{1j}(t);$$

$$i = 1, 2, \dots, m_1, \quad j = 1, 2, \dots, m_2, \quad (12)$$

$$\theta_{2i}(t+1) = \theta_{2i}(t) + \eta g'[nt_{2i}(t)] \delta(t) W_{3i}(t); \quad i = 1, 2, \dots, m_2. \quad (13)$$

在上述式子中, η 为学习步长, $g'(x) = \frac{1}{2} [1 - g^2(x)]$. 设神经网络反馈控制器的一般形式为

$$u(t) = h[e(t), e(t-1), \dots, e(t-p)]. \quad (14)$$

其中 $e(t)$ 为系统给定 $r(t)$ 与系统输出 $y(t)$ 之间的误差

$$e(t) = r(t) - y(t). \quad (15)$$

我们亦用一个三层 BP 网络来构成控制器, 即

输入层:
$$c_{1i}(t) = e(t-i+1); \quad i = 1, 2, \dots, n_1. \quad (16)$$

其中 $n_1 = p$.

隐层:
$$s_{2i}(t) = \sum_{j=1}^{n_1} k_{2ij} c_{1j}(t) + \beta_{2i}; \quad i = 1, 2, \dots, n_2, \quad (17)$$

$$c_{2i}(t) = v[s_{2i}(t)]; \quad i = 1, 2, \dots, n_2. \quad (18)$$

输出层:
$$s_3(t) = \sum_{j=1}^{n_2} k_{3j} c_{2j}(t) + \beta_3, \quad (19)$$

$$u(t) = v[s_3(t)]. \quad (20)$$

在上述式子中, k_{2ij}, k_{3j} 为权系数, β_{2i}, β_3 为阈值, $v(x) = \frac{2}{1 + e^{-x}} - 1$ 为激励函数. 为确定神经

网络控制器的权系数, 本文用神经网络模型输出 $\hat{y}(t+1)$ 来近似 $y(t+1)$, 从而用 $\frac{\partial \hat{y}}{\partial u}$ 来近似

specialised learning 算法中的 $\frac{\partial y}{\partial u}$. 据文献[2], 经有限次适当学习后, \hat{y} 能准确地逼近 y . 因此

这样的修正是可行的, 设性能指标:

$$J_2 = \frac{1}{2} [r(t+1) - \hat{y}(t+1)]^2. \quad (21)$$

应用 specialised learning 算法, 可得如下权修正公式:

$$k_{3i}(t+1) = k_{3i}(t) + \lambda[r(t+1) - \hat{y}(t+1)]v'[s_3(t)]c_{2i}(t) \frac{\partial \hat{y}(t+1)}{\partial u(t)};$$

$$i = 1, 2, \dots, n_2, \quad (22)$$

$$\beta_{3i}(t+1) = \beta_{3i}(t) + \lambda[r(t+1) - \hat{y}(t+1)]v'[s_3(t)] \frac{\partial \hat{y}(t+1)}{\partial u(t)}, \quad (23)$$

$$k_{2ij}(t+1) = k_{2ij}(t) + \lambda[r(t+1) - \hat{y}(t+1)]v'[s_3(t)]k_{3i}(t)v'[s_{2i}(t)]c_{1j}(t) \frac{\partial \hat{y}(t+1)}{\partial u(t)};$$

$$i = 1, 2, \dots, n_2; \quad j = 1, 2, \dots, n_1. \quad (24)$$

$$\beta_{2i}(t+1) = \beta_{2i}(t) + \lambda[r(t+1) - \hat{y}(t+1)]v'[s_3(t)]k_{3i}(t)v'[s_{2i}(t)] \frac{\partial \hat{y}(t+1)}{\partial u(t)};$$

$$i = 1, 2, \dots, n_2. \quad (25)$$

上述式子中, $v'(x) = \frac{1}{2}[1 - v^2(x)]$, λ 为学习步长, $\frac{\partial \hat{y}}{\partial u}$ 可由(2)~(6)式求得, 即

$$\begin{aligned} \frac{\partial \hat{y}(t+1)}{\partial u(t)} &= \frac{\partial \hat{y}(t+1)}{\partial nt_3(t)} \sum_{i=1}^{m_2} \frac{\partial nt_3(t)}{\partial o_{2i}(t)} \cdot \frac{\partial o_{2i}(t)}{\partial nt_{2i}(t)} \cdot \frac{\partial nt_{2i}(t)}{\partial o_{1n+1}(t)} \\ &= g'[nt_3(t)] \sum_{i=1}^{m_2} w_{3i} w_{2,i,n+1} g'[nt_{2i}(t)]. \end{aligned} \quad (26)$$

由此我们可得到如下自适应控制算法:

- 1) 采样得到 $y(t), r(t)$;
- 2) 将 $\{e(t), \dots, e(t-p)\}$ 输入神经网络控制器得到 $u(t)$;
- 3) 将 $u(t)$ 输入对象及神经网络模型;
- 4) 用(10)~(13)式修正模型的权及阈值;
- 5) 用(22)~(26)式修正控制器的权及阈值;
- 6) 返回 1), 继续进行.

3 基于 BP 网络的自适应 PID 控制

在前述基于 BP 网络的自适应控制的基础上, 通过适当简化, 得到 BP 网络自适应 PID 控制器. 令神经网络控制器为具有两层结构的网络, 其输入层有三个节点, 其输入分别为

$$o_1(t) = e(t); \quad o_2(t) = \sum_{i=1}^l e(i); \quad o_3(t) = e(t) - e(t-1),$$

其输出节点数为 1, 且令

$$u(t) = \sum_{i=1}^3 k_i(t) o_i(t). \quad (27)$$

由此可见, 控制器具有比例、积分和微分控制结构, 据上节的结果, 易求得控制器的权系数, 也即比例、积分和微分增益的在线修正公式:

$$k_i(t+1) = k_i(t) + \lambda[r(t+1) - \hat{y}(t+1)]o_i(t)g'[nt_3(t)] \sum_{i=1}^{m_2} w_{3i} w_{2,i,n+1} g'[nt_{2i}(t)]. \quad (28)$$

4 仿真实例

为验证本文方法的有效性, 进行了大量仿真实验, 现仅举两例说明.

例 1 被控对象为

$$y(t) = \frac{0.91y(t-1) + u(t-1)}{1 + y(t-2)u(t-2)}$$

用于辨识的 BP 网络模型结构为 4-6-1, $\eta=0.65$. BP 网络控制器结构为 3-4-1, $\lambda=0.65$. 图 1 所示的为 BP 网络自适应控制响应, 可见其结果令人满意.

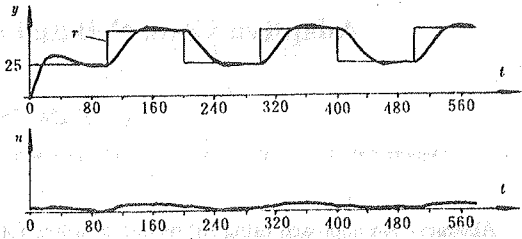


图 1 BP 网络自适应控制响应

例 2 被控对象为时变非线性系统

$$y(t) = \begin{cases} \frac{0.91y(t-1) + u(t-1)}{1 + y^2(t-1)}, & t \leq 200, \\ \frac{y(t-1) + 1.2u(t-1) + 0.25y(t-1)u(t-1)}{1 + y^2(t-1)}, & t > 200. \end{cases}$$

采用本文提出的 BP 网络自适应 PID 控制方法. 其中神经网络模型结构为 4-6-1, $\eta=0.65$. 控制器的学习步长 $\lambda=0.65$. 图 2 (a) 为系统响应 y (实线表示) 及模型输出 \hat{y} (虚线表示) 的曲线; 图 2 (b) 为控制量 u 变化曲线. 从图 2 可知, BP 网络自适应 PID 控制有较强的适应性和鲁棒性. 此外, 模型的跟踪能力也令人满意.

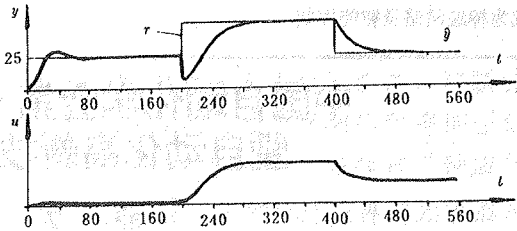


图 2 BP 网络自适应 PID 控制响应

5 结束语

本文对文献[1]提出的 specialised learning 方法进行了改进, 由此得到 BP 网络自适应控制及 BP 网络自适应 PID 控制方法. 这些方法的特点为:

- 1) 可以在线进行学习和控制;
- 2) 能有效地对付一些较复杂非线性被控对象, 特别是难于用传统方法建模的被控对象;
- 3) 有较强的鲁棒性和适应性.

此外, 本文提出的 BP 网络自适应 PID 控制器的结构简单, 易于实现, 将神经网络与 PID 控制结合, 将使 PID 控制器具备对付更复杂的非线性系统的能力.

参 考 文 献

- [1] Psaltis, D. and Yamamura, A.. A Multilayered Neural Network Controller. IEEE Control Syst. Mag, 1988, 8(2):17-21
- [2] Hecht-Nielsen, R.. Theory of the Backpropagation Neural Network. IJCNN'89, 1989, 1:593-606
- [3] Saeren, M. and Soquet, A.. A Neural Controller. 1st IEE Int. Conf on Artificial Neural Networks, 1989, 211-215

Adaptive Control Based on BP Neural Networks

TAN Yonghong

(Department of Computer Engineering, Guilin Institute of Electronic Technology • Guilin, 541004, PRC)

Abstract: An approach using BP neural networks for on-line process identification and control is presented in this paper. A modification of the specialised learning method for the neural adaptive control is developed. Furthermore, the paper proposes an adaptive PID controller based on the proposed neural adaptive control scheme.

Key words: neural networks; adaptive control; learning

本文作者简介

谭永红 1958年生。1988年华南理工大学自动化系获硕士学位,现为桂林电子工业学院计算机系任副教授。研究领域为神经网络及智能控制。

中国自动化学会第十届青年学术年会 暨自动化高新技术产品展示会

征 文 通 知

根据中国自动化学会 [1993] 自学字第 039 号文件,中国自动化学会第十届青年学术年会(CYA'94)将于 1994 年 8 月下旬在著名古城西安召开。这是一次促进青年科技工作者迅速成长,促进自动控制及相关学科青年学者学术交流的盛会,届时将邀请国内著名专家学者作综述或专题报告,组织专题讨论,评选优秀论文,同时还将举办自动化高新技术产品展示会。

会议主题: 自动化理论、技术及应用

征文范围: 1) 大系统、非线性系统、复杂大系统控制; 2) 自调节、自适应控制、变结构控制; 3) 智能控制、模糊控制、专家系统、神经网络系统; 4) 线性系统、广义系统; 5) 镇定与鲁棒控制、预测控制、最优控制理论及应用、 H_2 控制; 6) 辨识、估计,故障诊断与容错控制; 7) 传感器与测试技术、建模与仿真、数字信号处理; 8) 过程控制与计算机控制系统; 9) 机器人控制、仿真、视觉; 10) 并行分布处理、信息融合理论与应用; 11) 分布参数系统、离散事件系统、混合动态系统理论; 12) 空间及运动体控制; 13) 系统工程、宏观经济管理、控制与决策; 14) 自动控制教学与实验; 15) 其它相关领域的新理论、新方法、新技术。

来稿要求: 1) 理论联系实际,内容扎实具体,具有学术价值。2) 未在国内公开发行的刊物或全国性学术会议上发表或宣读过。3) 来稿请注明论文所属征文范围中的类别号。4) 为了保证论文集出版质量以及稿件录用后及时交稿,并减少作者的重复劳动,要求来稿论文按《控制与决策》、《信息与控制》写作格式用计算机打印,每篇论文以 5 页为限(包括图、文)。

论文截止日期: 1) 1994 年 5 月 1 日前提交两份论文全文初稿。2) 1994 年 5 月 30 日前发出论文录取通知。3) 1994 年 6 月 20 日前提交正式论文全文。

投稿地址: 710072 西安市西北工业大学自动控制系 613 信箱

联系人: 张友民、丁振 电话: 5253371~5253379-2814

欢迎广大青年科技工作者积极投稿,来稿请写清作者的年龄,详细通讯地址及邮政编码。并在信封上注明“CYA'94”字样。请作者自留底稿,来稿恕不退回。

中国自动化学会第十届青年学术年会组委会