

# 基于 BP 神经网络整定的 PID 控制及其仿真

高富强<sup>1</sup>, 李 萍<sup>2</sup>, 张磊敏<sup>3</sup>, 曾令可<sup>4</sup>, 涂 腾<sup>5</sup>, 刁浩明<sup>5</sup>

(1. 广州能源检测研究院, 广州 511447; 2. 中国科学院广州能源研究所, 广州 510640;  
3. 江门出入境检验检疫局, 江门 529000; 4. 华南理工大学, 广州 510640;  
5. 广州繁诺节能科技有限公司, 广州 510000)

**摘 要** PID 控制是连续系统控制理论中技术成熟、应用广泛的一种控制方法。但在实际应用中, 其参数的整定往往是依靠经验和现场调试。本文把 BP(Back Propagation)神经网络技术应用到 PID 控制中, 利用其非线性函数逼近能力, 对 PID 控制器进行整定, 并通过仿真试验取得较好的结果。

**关键词** PID; 整定; BP 神经网络; 仿真

中图分类号: TQ174.6+58

文献标识码: A

## 前言

对于经典的 PID 控制器, 要取得较理想的控制效果, 必须先整定好三个参数: 比例系数 ( $K_p$ )、积分时间 ( $T_i$ )、微分时间 ( $T_d$ )。在实际应用中, 不同的对象对控制器的特性有不同的要求, 整定的目的就是设法使控制器的特性和被控对象配合好, 以便得到最佳的控制效果。

控制器的特性与被控对象之间存在着既相互配合又相互制约的关系, 这种关系往往是复杂的“非线性组合”。BP(Back Propagation)神经网络

具有任意非线性表达能力, 可以通过对系统性能的学习来实现具有最佳组合的 PID 控制器。

## 1 经典的 PID 控制器

### 1.1 经典的 PID 控制系统框图



图 1 经典的 PID 控制系统框图

### 1.2 经典的 PID 控制算法

$$\begin{aligned}
 u(k) &= u(k-1) + \Delta u(k) \\
 \Delta u(k) &= k_p \cdot [\text{error}(k) - \text{error}(k-1)] + k_i \cdot \text{error}(k) + \\
 &\quad k_d \cdot [\text{error}(k) - 2\text{error}(k-1) + \text{error}(k-2)]
 \end{aligned}
 \tag{1}$$

式中:  $u(k)$  —— 第  $k$  次采样时控制器的输出量;  
 $\text{error}(k)$  —— 第  $k$  次采样时设定值与实际值的偏差量;

$k_p$  —— 比例放大系数;

$k_i$  —— 积分系数;

$k_d$  —— 微分系数;

其中,  $k_p$ 、 $k_i$ 、 $k_d$  都是常数, 这三个常数需要依靠经验和现场调试来确定。

## 2 BP 神经网络

### 2.1 BP 算法

BP 算法(反向传播算法)是一种监督式的学

基金项目: 广东省应用性科技研发专项资金项目(项目号 2015B020238001)

广东省质量技术监督局科技项目(项目号 2016PJ016)

收稿日期: 2017-05-15

作者简介: 高富强, (1986—), 工学硕士, 计量工程师。研究方向: 能源计量、节能减排。

习算法。它是通过连续不断地在相对于误差函数斜率下降的方向上计算网络权值和偏差的变化而逐渐逼近目标的。每一次权值和偏差的变化都与网络误差的影响成正比,并以反向传播的方式传递到每一层。

BP 算法是由两部分组成:信息的正向传播与误差的反向传播。在正向传播过程中,输入信息从输入经隐含层逐层计算传向输出层,每一层神经元的状态只影响下一层神经元的状态。如果在输出层没有得到期望的输出,则计算输出层的误差变化值,然后转向反向传播,通过网络将误差信号沿原来的连接通路反传回来修改各层神经元的权值直至达到期望目标。

### 2.2 BP 神经网络

BP 神经网络的产生归功于 BP 算法的获得。它有一个输入层、一个输出层和一个或多个隐含层,同层神经元间无关联,异层神经元间向前连接。根据对象的复杂程度,选择适当的网络结构,就可以实现从输入空间到输出空间的任意非线性函数的映射。

BP 神经网络主要用于:① 函数逼近:用输入矢量和相应的输出矢量训练一个网络逼近一个函数;② 模式识别:用一个特定的输出矢量将它与输入矢量联系起来;③ 分类:把输入矢量以所定义的合适方式进行分类;④ 数据压缩:减少输出矢量维数以便于传输或存储。

## 3 基于 BP 神经网络整定的 PID 控制

### 3.1 基于 BP 神经网络的 PID 控制器结构

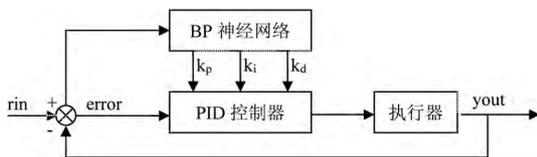


图 2 基于 BP 神经网络的 PID 控制系统框图

$$net_i^{(3)}(k) = \sum_{i=1}^Q w_{li}^{(3)}(k) \cdot O_i^{(2)}(k)$$

$$O_i^{(3)}(k) = g(net_i^{(3)}(k)) \quad (l=1,2,3) \quad (5)$$

$$O_1^{(3)}(k) = k_p(k) O_2^{(3)}(k) = k_i(k) O_3^{(3)}(k) = k_d(k)$$

输出层输出节点分别对应三个 PID 控制参

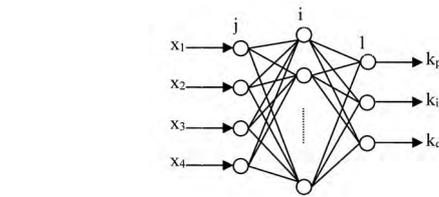


图 3 BP 神经网络结构

基于 BP 神经网络的 PID 控制器由两部分组成:① 经典的 PID 控制器;② BP 神经网络。BP 神经网络利用误差反向传播特性,把控制偏差传回到网络的各层神经元,根据 BP 算法计算各层神经元的权值,输出 PID 控制参数给 PID 控制器,经过不断修正自身网络权值,最终使控制偏差趋于零。

### 3.2 BP 神经网络的设计

采用三层 BP 神经网络,其结构如图 3 所示。

网络输入层的输入为:

$$O_j^{(1)}(k) = x(j)(k) \quad (j=1,2,\dots,M) \quad (2)$$

式中,输入变量的个数 M 取决于被控系统的复杂程度。

网络隐含层的输入、输出为:

$$net_i^{(2)}(k) = \sum_{j=1}^M w_{ij}^{(2)}(k) \cdot O_j^{(1)}(k)$$

$$O_i^{(2)}(k) = f(net_i^{(2)}(k))$$

$$(i=1,2,\dots,Q) \quad (3)$$

式中,  $w_{ij}^{(2)}(k)$  为隐含层加权系数;上角标 (1)、(2)、(3) 分别代表输入层、隐含层和输出层。

隐含层神经元的活化函数取正负对称的 Sigmoid 函数:

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (4)$$

网络输出层的输入、输出为:

数。由于  $k_p, k_i, k_d$  不能为负值,所以输出层神经

元的活化函数取非负的 Sigmoid 函数:

$$g(x) = \frac{1}{2} [1 + \tanh(x)] = \frac{e^x}{e^x + e^{-x}} \quad (6)$$

取性能指标函数为:

$$E(k) = \frac{1}{2} [\text{rin}(k) - \text{yout}(k)]^2 \quad (7)$$

根据 BP 算法修正网络权值,即按  $E(k)$  对加权系数的负梯度方向搜索调整,并附加一个使搜索快速收敛全局极小的惯性项:

$$\Delta w_{ii}^{(3)}(k) = -\eta \cdot \frac{\partial E(k)}{\partial w_{ii}^{(3)}(k)} + \alpha \cdot \Delta w_{ii}^{(3)}(k-1) \quad (8)$$

式中:  $\eta$  为学习速率;  $\alpha$  为惯性系数。

$$\frac{\partial E(k)}{\partial w_{ii}^{(3)}(k)} = \frac{\partial E(k)}{\partial y(k)} \cdot \frac{\partial y(k)}{\partial \Delta u(k)} \cdot \frac{\partial \Delta u(k)}{\partial O_i^{(3)}(k)} \cdot \frac{\partial O_i^{(3)}(k)}{\partial \text{net}_i^{(3)}(k)} \cdot \frac{\partial \text{net}_i^{(3)}(k)}{\partial w_{ii}^{(3)}(k)} \quad (9)$$

$$\frac{\partial \text{net}_i^{(3)}(k)}{\partial w_{ii}^{(3)}(k)} = O_i^{(2)}(k) \quad (10)$$

由于  $\frac{\partial y(k)}{\partial \Delta u(k)}$  未知,所以近似用符号函数  $\text{sgn}(\frac{\partial y(k)}{\partial \Delta u(k)})$  取代,又由于分母不能为零,所以用  $\text{sgn}\{[y(k) - y(k-1)] \cdot [\Delta u(k) - \Delta u(k-1)]\}$  代替。由此带来计算不精确的影响可以通过调整学习速率  $\eta$  来补偿。

由式(1)和式(5)可求得:

$$\frac{\partial \Delta u(k)}{\partial O_1^{(3)}(k)} = \text{error}(k) - \text{error}(k-1) \quad (11)$$

$$\frac{\partial \Delta u(k)}{\partial O_2^{(3)}(k)} = \text{error}(k) \quad (12)$$

$$\frac{\partial \Delta u(k)}{\partial O_3^{(3)}(k)} = \text{error}(k) - 2\text{error}(k-1) + \text{error}(k-2) \quad (13)$$

上述分析可得网络输出层权值的学习算法为:

$$\text{yout}(k) = \frac{\epsilon(k) \cdot [11 \cdot e^{\text{yout}(k-1)} + 3 \cdot e^{-\text{yout}(k-1)}] \cdot \text{yout}(k-1)}{12 \cdot e^{\text{yout}(k-1)}} + u(k-1)$$

式中,系数  $\epsilon(k)$  是慢时变的,  $\epsilon(k) = 1.2 \cdot (1 - 0.8 \cdot e^{-0.1k})$ 。

神经网络的结构选择 4-5-3,其中输入层  $O^{(1)}(k) = [\text{rin}(k), \text{yout}(k), \text{error}(k), 1]$ , 学

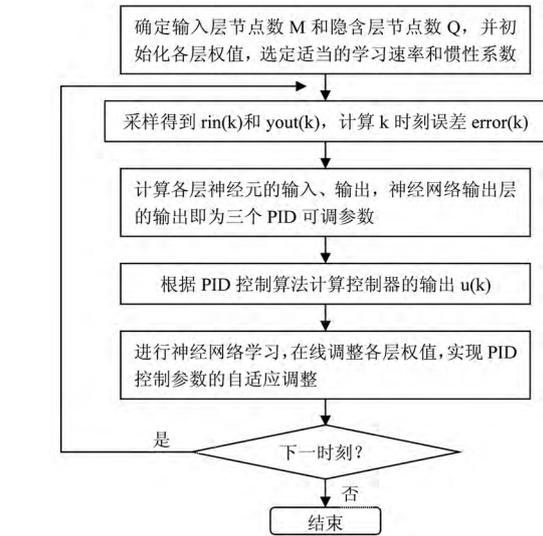


图 4 BP 神经网络整定的 PID 控制算法步骤

$$\Delta w_{ii}^{(3)}(k) = \eta \cdot \delta_i^{(3)}(k) \cdot O_i^{(2)}(k) + \alpha \cdot \Delta w_{ii}^{(3)}(k-1) \quad (14)$$

其中:  $\delta_i^{(3)}(k) = \text{error}(k) \cdot \text{sgn}\{[y(k) - y(k-1)] \cdot [\Delta u(k) - \Delta u(k-1)]\} \cdot \frac{\partial \Delta u(k)}{\partial O_i^{(3)}(k)} \cdot g'(\text{net}_i^{(3)}(k))$

同理可得隐含层权值的学习算法:

$$\Delta w_{ij}^{(2)}(k) = \eta \cdot \delta_i^{(2)}(k) \cdot O_j^{(1)}(k) + \alpha \cdot \Delta w_{ij}^{(2)}(k-1) \quad (15)$$

其中:  $\delta_i^{(2)}(k) = f'(\text{net}_i^{(2)}(k)) \cdot \sum_{l=1}^3 \delta_l^{(3)}(k) \cdot w_{li}^{(3)}(k)$

式中,  $g'(\cdot) = 2 \cdot g(x) \cdot [1 - g(x)]$ ,  $f'(\cdot) = 1 - f^2(x)$ 。

### 3.3 基于 BP 神经网络整定的 PID 控制算法步骤

## 4 仿真试验

设被控对象的近似数学模型为:

学习速率  $\eta = 0.001$  和惯性系数  $\alpha = 0.03$ , 各层权值初始值取区间  $[-0.5, 0.5]$  上的随机数。设定输入为:

$$rin(k) = \begin{cases} \sin(\pi k/150) & k < 1500 \\ 0.5 & 1500 \leq k < 3000 \\ 1 & 3000 \leq k < 4500 \\ 0.6\sin(\pi k/150) + 0.8\sin(\pi k/200) + 0.6\sin(\pi k/250) & k \leq 6000 \end{cases}$$

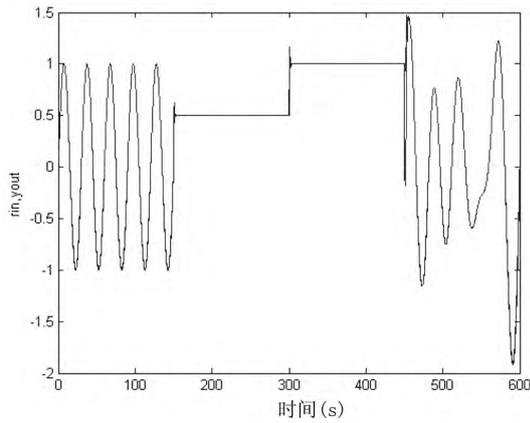


图 5 基于 BP 神经网络整定的 PID 控制 rin(k) 与 yout(k) 曲线

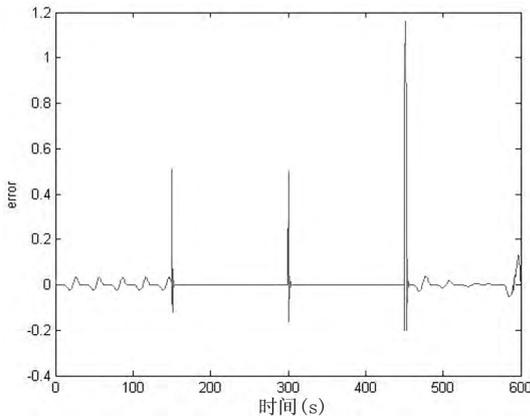


图 6 跟踪误差曲线

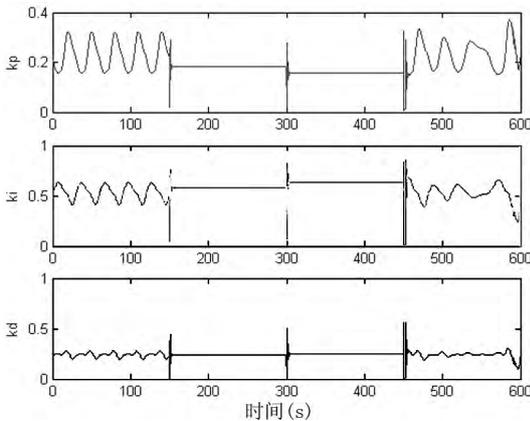


图 7 参数自适应整定曲线

本试验采用 Matlab6.5 对其进行仿真, 仿真结果如图 5~图 7 所示。由图可以看出, 尽管对象的设定输入 rin(k) 多变化, 但是被控对象仍能较好的跟踪设定输入。

### 5 结束语

神经网络具有自学习、自组织能力, 可以实现从输入空间到输出空间的任意非线性函数的映射。利用 BP 神经网络的这种性能对 PID 控制参数进行在线自整定, 仿真结果表明其控制效果较令人满意。

### 参考文献

- [1] 刘金琨. 先进 PID 控制及 Matlab 仿真[M]. 北京: 电子工业出版社. 2003.
- [2] 曾令可, 叶卫平. 计算机在材料科学与工程中的应用[M]. 武汉: 武汉理工大学出版社. 2004.
- [3] 余阳春, 王晓春. FUZZY-PID 控制系统在工业窑炉控制中的应用[J]. 陶瓷学报. 2006, 27(3): 339-342.
- [4] 廖芳芳, 肖建. 基于 BP 神经网络 PID 参数自整定的研究[J]. 系统仿真学报. 2005, 17(7): 1711-1713.
- [5] 李萍, 曾令可, 等. 基于 MATLAB 的 BP 神经网络预测系统的设计[J]. 计算机应用与软件, 2008, 25(4): 149-150.
- [6] 李颖冲, 陈小强, 王思明, 等. 基于模糊自适应 PID 的陶瓷窑炉温度控制系统仿真[J]. 中国陶瓷, 2009, 47(7): 51-54.
- [7] 朱庆霞, 胡国林, 唐奇, 等. 液化气梭式窑能耗的计算机仿真研究[J]. 陶瓷学报. 2005, 26(1): 24-28.

## PID Controllers Based on BP Neural Networks and the Simulation

Gao Fuqiang<sup>1</sup>, Li Ping<sup>2</sup>, Zhang Leimin<sup>3</sup>, Zeng Lingke<sup>4</sup>, Tu Teng<sup>5</sup>, Diao Haoming<sup>5</sup>

(1. Guangzhou Institute of Energy Testing, Guangzhou Guangdong 510640, China;

2. Guangzhou Institute of Energy Conversion, Chinese Academy of Sciences, Guangzhou Guangdong 510640, China;

3. Inspection & Quarantine Technical Center of Jiangmen CIQ, Jiangmen Guangdong 529000, China;

4. South China University of Technology, Guangzhou Guangdong 510640, China;

5. Guangzhou Famuo Energy Saving Technology Co., Ltd., Guangdong, China)

**Abstract:** PID controllers are widely used in cybernetics of continuous system, and the technique is mature. But in practice, the tuning of PID parameters depends on experience and debug. In this paper, BPNN (Back Propagation Neural Network) technique was used to tune PID parameters because of its non-linear function capability. The simulation proves it is satisfactory.

**Key words:** PID; Tuning; Back Propagation Neural Network; Simulation

\*\*\*\*\*

## 科学家用纳米技术研制出陶瓷“海绵”

陶瓷通常又硬又脆,跟海绵完全扯不上关系。美国和中国科学家最近合作,利用纳米纤维制造出一种陶瓷“海绵”,它可压缩变形,并且能耐高温、隔热性好,有着广泛的潜在用途。

美国布朗大学日前发布的新闻公报说,这一成果是由布朗大学与中国清华大学合作取得的,可望低成本、大规模生产纳米陶瓷“海绵”,用于制造新型隔热材料、净水材料等。相关论文发表在美国《科学》杂志的子刊《科学进展》上。

普通陶瓷上的裂纹会迅速扩大,轻微变形就可能使整块陶瓷破碎。但在纳米尺度上,材料特性会发生巨大变化,以陶瓷纳米纤维制成的物品可以很柔韧。不过,目前制造纳米纤维的主要方法要么不适用于陶瓷,要么成本高昂而且费时间。

研究小组采用了一种被称为“溶液喷射纺丝”的技术,利用气体压力使包含陶瓷材料的溶液从极细的针孔中喷出,凝固成纳米尺度的丝线。把这些丝线收集起来加热,去除溶剂材料,剩下的就是绕成一团的陶瓷纤维,外观看上去像棉球。

研究人员用多种不同类型的陶瓷材料成功制成“海绵”,试验发现它们的可压缩性很好,能耐 800℃ 高温,而且高度隔热。把花瓣放在 7mm 厚的氧化锆陶瓷海绵上,下面以 400℃ 加热 10 分钟,花瓣仍完好无损,而用普通陶瓷材料隔热,花瓣早就烧焦了。