

# 一种基于混合梯度下降算法的模糊神经网络设计及应用

韩红桂<sup>†</sup>, 林征来, 乔俊飞

(1. 北京工业大学 电子信息与控制工程学院, 北京 100124;

2. 北京工业大学 计算智能与智能系统北京市重点实验室, 北京 100124)

**摘要:** 为了提高模糊神经网络(FNN)的收敛速度和泛化能力,提出一种基于混合梯度下降算法(HG)的模糊神经网络(HG-FNN). HG-FNN通过设计FNN参数调整过程的自适应学习率,利用链式法则获取FNN参数学习过程的梯度,在实现FNN参数自校正的同时,给出HG-FNN的收敛性证明,保证HG-FNN的收敛速度和泛化能力.最后,将所设计的HG-FNN应用于非线性系统建模与污水处理过程关键水质参数预测,实验比较结果显示,HG-FNN不仅具有较快的收敛速度,而且具有较好的泛化能力.

**关键词:** 模糊神经网络; 混合梯度; 自适应学习率; 非线性系统建模

中图分类号: TP173

文献标志码: A

## Design and application of hybrid gradient descent-based fuzzy neural network

HAN Hong-gui<sup>†</sup>, LIN Zheng-lai, QIAO Jun-fei

(1. College of Electronic Information & Control Engineering, Beijing University of Technology, Beijing 100124, China;

2. Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing University of Technology, Beijing 100124, China)

**Abstract:** To improve the convergence speed and generalization ability of the fuzzy neural network(FNN), a fuzzy neural network, based on the hybrid gradient(HG) descent algorithm, is proposed in this paper. This HG-FNN can obtain the adaptive learning rate of the parameter adjustment process. Then, the chain rule is used to calculate the gradient descent of the learning process to adjust the parameters of FNN. Meanwhile, the convergence proof of HG-FNN is given in details to ensure the convergence speed and the precision of FNN. Finally, the proposed HG-FNN is used to model the nonlinear systems and predict the effluent qualities of wastewater treatment process. The results show that the proposed HG-FNN owns faster convergence speed, as well as with suitable generalization ability than other FNNs.

**Keywords:** fuzzy neural network; hybrid gradient; adaptive learning rate; nonlinear systems modeling

## 0 引言

模糊神经网络(FNN)具有强大的推理和学习能力,已经在非线性系统辨识、控制等领域得到了广泛应用<sup>[1-4]</sup>.然而,FNN应用的关键问题之一是其参数学习问题<sup>[5-7]</sup>.

围绕FNN的参数学习的问题,国内外学者相继提出一些学习算法,目前常用的主要有以下几类:

1) 误差反向传播(BP)算法.该算法由于具有简单有效的特点使其最为常用<sup>[8-10]</sup>.为了提高网络性能,Zhang等<sup>[11]</sup>提出了一种基于动态学习率的在线BP全局收敛性算法,该算法通过误差函数的下降率

作为参数调整的学习率,并给出在特定条件下全局收敛的理论证明,但是方法中参数太多,设置较复杂.Han等<sup>[12]</sup>提出了一种基于敏感度分析算法(SA)生长剪枝模糊神经网络(GP-FNN)的参数优化方法,该方法利用敏感度输出作为梯度下降算法的自适应学习策略,从而对FNN的中心和宽度进行调整,但是网络精度有待提高.Davanipoor等<sup>[13]</sup>提出一种加速混合的模糊小波网络(FWNN)参数优化方法,该方法给出了一种局部自适应学习率,使其网络中每一参数都有一个自适应学习率,从而保证网络预测精度,但是方法计算代价大,容易出现病态的情况.BP算法的

收稿日期: 2016-08-13; 修回日期: 2017-01-10.

基金项目: 国家自然科学基金项目(61533002,61622301); 北京市自然科学基金项目(4172005); 科技部水专项(2017ZX07104).

作者简介: 韩红桂(1983-),男,教授,博士生导师,从事污水处理过程建模、优化与控制等研究;林征来(1991-),男,硕士生,从事复杂过程建模、控制的研究.

<sup>†</sup>通讯作者. E-mail: recharhan@bjut.edu.cn

原理是根据梯度下降最小化目标函数,具有运算时间过长、收敛速度过慢的问题。

2) 最小二乘法。为提高收敛速度,学者们提出了具有极快收敛速度的最小二乘法。例如,Wang等<sup>[14]</sup>提出了一种基于扩展卡尔曼滤波(EKF)算法的在线自组织模糊神经网络(FAOS-PFNN)参数优化方法,该方法利用EKF对FNN参数进行调整,从而保证网络收敛性能。但是,EKF算法的初始状态不容易确定,训练过程中会出现震荡的现象,容易导致滤波器发散以致网络精度难以保证的问题。Rubio<sup>[15]</sup>提出了一种在线自组织改进型模糊最小二乘法(SOFMLS),该方法能够实现FNN参数自校正,网络性能有所提高。但是,当网络参数维数较高时运行速度会变慢,而且有时会出现病态运算使得网络泛化能力大大降低。最小二乘法虽然具有较快的收敛速度,但是容易陷入局部极小的问题没有得到解决。

3) 进化算法。为了解决局部极小值的问题,学者们提出了具有较好鲁棒性和全局搜索能力的进化算法。Leng等<sup>[16]</sup>提出了一种基于遗传算法(GA)的新颖混杂自组织模糊神经网络(SOFNNGA)参数优化算法,该方法利用GA对FNN规则的隶属函数宽度等参数进行调整,SOFNNGA不依赖于现有推理,使用范围更加广泛。Mashinchi等<sup>[17]</sup>提出了基于GA的两阶段学习方法,首先粗略估计FNN最优模糊权重,然后精确估计FNN隶属函数的形状,从而提高整个模糊神经网络的性能,但是GA是一种全局优化算法,涉及大量的个体计算,其收敛速度慢,时间消耗代价很高。Kuo等<sup>[18]</sup>提出了一种基于PSO的模糊神经网络(IOAP-FNN)参数优化算法,该方法利用粒子群优化(PSO)对FNN的结构和参数进行调整从而寻求网络最佳性能。但是,由于PSO是全局搜索算法,在很大程度上降低了其整体学习速度,并且没有理论的收敛性证明,限制了其推广应用。虽然进化算法具有全局搜索的能力,但存在搜索时间过长,收敛速度慢的问题。因此,对FNN算法的设计仍需进一步研究。

鉴于以上问题,本文提出一种基于混合梯度下降算法的模糊神经网络(HG-FNN)。该HG-FNN通过设计FNN参数调整过程的自适应学习率,利用链式法则完成对FNN参数梯度的求解。并给出了HG-FNN的收敛性证明,保证FNN的收敛速度和泛化能力。所提出的混合梯度下降算法的模糊神经网络能够实现参数自校正,具有良好的自适应能力。利用其分别对非线性系统建模与污水处理关键参数建模,验证了该方法的有效性。

## 1 基于混合梯度下降算法的模糊神经网络(HG-FNN)

### 1.1 模糊神经网络(FNN)

FNN主要由输入层、RBF层、规则化层和输出层组成。FNN(多输入单输出)的输出可描述为

$$y = \sum_{l=1}^p w_l v_l = \frac{\sum_{l=1}^p w_l e^{-\sum_{i=1}^m \frac{u_i - c_{il}}{2\sigma_{il}^2}}}{\sum_{j=1}^p e^{-\sum_{i=1}^m \frac{u_i - c_{ij}}{2\sigma_{ij}^2}}}, \quad (1)$$

$$\beta_l = w_l v_l = \frac{w_l e^{-\sum_{i=1}^m \frac{u_i - c_{il}}{2\sigma_{il}^2}}}{\sum_{j=1}^p e^{-\sum_{i=1}^m \frac{u_i - c_{ij}}{2\sigma_{ij}^2}}}. \quad (2)$$

其中: $p$ 为规则化层神经元数; $j = 1, 2, \dots, p$ ;  $l = 1, 2, \dots, p$ ;  $\mathbf{w} = [w_1, w_2, \dots, w_p]$ 为规则化层与输出层之间的连接权值; $\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_p]$ 为输出层的输入; $\beta_l$ 为规则化层的第 $l$ 个神经元的输出; $y$ 为模糊神经网络的输出。

$$v_l = \frac{\varphi_l}{\sum_{j=1}^p \varphi_j} = \frac{e^{-\sum_{i=1}^m \frac{u_i - c_{il}}{2\sigma_{il}^2}}}{\sum_{j=1}^p e^{-\sum_{i=1}^m \frac{u_i - c_{ij}}{2\sigma_{ij}^2}}}. \quad (3)$$

其中: $\mathbf{v} = [v_1, v_2, \dots, v_p]^T$ 为规则化层输出, $v_l$ 为第 $l$ 个神经元的输出。

$$\varphi_j = \prod_{i=1}^m e^{-\frac{u_i - c_{ij}}{2\sigma_{ij}^2}} = e^{-\sum_{i=1}^m \frac{u_i - c_{ij}}{2\sigma_{ij}^2}}, \quad (4)$$

$$u_i = x_i. \quad (5)$$

其中: $i = 1, 2, \dots, m$ ;  $u_i$ 为第 $i$ 个神经元的输出; $\varphi_j$ 为第 $j$ 个神经元的输出; $c_{ij}$ 为神经元 $j$ 的中心值; $\sigma_{ij}$ 为神经元 $j$ 的隶属函数宽度; $\mathbf{x} = [x_1, x_2, \dots, x_m]$ 。

### 1.2 混合梯度下降算法(HG)

为了提高FNN性能,提出一种混合梯度下降算法用来调整模糊神经网络的中心 $\mathbf{c}$ 、宽度 $\boldsymbol{\sigma}$ 、输出权值 $\mathbf{w}$ 参数。定义目标函数 $E$ 如下:

$$E(t) = \frac{1}{2}(y_d(t) - y(t))^2. \quad (6)$$

其中: $y_d(t)$ 为 $t$ 时刻期望输出, $y(t)$ 为 $t$ 时刻网络输出。

假设 $N$ 为1,有

$$E(t) = \frac{1}{2}(y_d(1) - y(1))^2. \quad (7)$$

传统的BP算法参数调整为

$$\begin{aligned} \Delta \boldsymbol{\theta}(t) &= -\eta \frac{\partial E(t)}{\partial \boldsymbol{\theta}(t)} = \\ &= -\eta (y_d(t) - y(t)) \frac{\partial y(t)}{\partial \boldsymbol{\theta}(t)} = -\eta e(t) \frac{\partial y(t)}{\partial \boldsymbol{\theta}(t)}. \end{aligned} \quad (8)$$

令

$$y'(\theta(t)) = \frac{\partial y(t)}{\partial \theta(t)}, \quad (9)$$

参数调整后,  $y(\theta(t+1))$  的一阶泰勒级数展开为

$$y(\theta(t+1)) = y(\theta(t)) + y'(\theta(t))\Delta\theta(t). \quad (10)$$

对于最优学习率  $\eta$ , 应使参数修正后接近期望值, 即

$$y_d(t) \approx y(\theta(t+1)) \approx y(\theta(t)) + y'(\theta(t))^T \Delta\theta(t). \quad (11)$$

将式(7)代入(10), 得到

$$y_d(t) - y(\theta(t)) \approx \eta(y_d(t) - y(\theta(t))) \|y'(\theta(t))\|^2. \quad (12)$$

式(11)两边同时乘以  $e(t)^2$ , 得到

$$\eta(t) = \frac{\|e(t)\|^2}{\|y'(\theta(t))e(t)\|^2} = \frac{\|e(t)\|^2}{\left\| \frac{\partial y(t)}{\partial \theta(t)} e(t) \right\|^2}. \quad (13)$$

令  $J(t) = \frac{\partial y(t)}{\partial \theta(t)}$ , 则  $\eta(t)$  为

$$\eta(t) = \frac{\|e(t)\|^2}{\|J(t)e(t)\|^2}. \quad (14)$$

FNN 参数  $c, \sigma, w$  更新规则如下:

$$c(t+1) = c(t) - \alpha\eta_1(t) \frac{\partial E(t)}{\partial c(t)} = c(t) - \alpha\eta_1(t) J_c(t)e(t), \quad (15)$$

$$\sigma(t+1) = \sigma(t) - \alpha\eta_1(t) \frac{\partial E(t)}{\partial \sigma(t)} = \sigma(t) - \alpha\eta_1(t) J_\sigma(t)e(t), \quad (16)$$

$$w(t+1) = w(t) - \gamma\eta_1(t) \frac{\partial E(t)}{\partial w(t)} = w(t) - \gamma\eta_1(t) J_w(t)e(t). \quad (17)$$

学习率  $\eta_1(t), \eta_2(t), \eta_3(t)$  分别为

$$\eta_1(t) = \frac{\|e(t)\|^2}{\|J_c(t)e(t)\|^2}, \quad (18)$$

$$\eta_2(t) = \frac{\|e(t)\|^2}{\|J_\sigma(t)e(t)\|^2}, \quad (19)$$

$$\eta_3(t) = \frac{\|e(t)\|^2}{\|J_w(t)e(t)\|^2}. \quad (20)$$

其中:  $\alpha, \gamma$  均为常数;  $\| \cdot \|$  为范数;  $\theta(t)$  包括3种变量, 分别为模糊神经网络的中心  $c$ 、宽度  $\sigma$ 、输出权值  $w$  参数, 当  $t=1$  时刻,  $\theta(1)$  表示为

$$\theta(1) = \begin{bmatrix} c_1(1) \cdots c_j(1) \cdots c_p(1) \\ \sigma_1(1) \cdots \sigma_j(1) \cdots \sigma_p(1) \\ w_1(1) \cdots w_j(1) \cdots w_p(1) \end{bmatrix}. \quad (21)$$

$t$  时刻雅克比矩阵  $j(t)$  分别表示为

$$j(t) = \begin{bmatrix} \frac{\partial e(t)}{\partial c_1(t)} \cdots \frac{\partial e(t)}{\partial c_j(t)} \cdots \frac{\partial e(t)}{\partial c_p(t)} \\ \frac{\partial e(t)}{\partial \sigma_1(t)} \cdots \frac{\partial e(t)}{\partial \sigma_j(t)} \cdots \frac{\partial e(t)}{\partial \sigma_p(t)} \\ \frac{\partial e(t)}{\partial w_1(t)} \cdots \frac{\partial e(t)}{\partial w_j(t)} \cdots \frac{\partial e(t)}{\partial w_p(t)} \end{bmatrix}. \quad (22)$$

根据链式法则, 雅克比矩阵  $j(t)$  的元素为

$$\frac{\partial e(t)}{\partial c_j(t)} = \left[ \frac{\partial e(t)}{\partial c_{1j}(t)} \quad \frac{\partial e(t)}{\partial c_{2j}(t)} \quad \cdots \quad \frac{\partial e(t)}{\partial c_{mj}(t)} \right], \quad (23)$$

$$\frac{\partial e(t)}{\partial c_{ij}(t)} = -\frac{2w_j(t) \times v_i(t) \times [x_i(t) - c_{ij}(t)]}{\sigma_{ij}(t)}, \quad (24)$$

$$\frac{\partial e(t)}{\partial \sigma_j(t)} = \left[ \frac{\partial e(t)}{\partial \sigma_{1j}(t)} \quad \frac{\partial e(t)}{\partial \sigma_{2j}(t)} \quad \cdots \quad \frac{\partial e(t)}{\partial \sigma_{mj}(t)} \right], \quad (25)$$

$$\frac{\partial e(t)}{\partial \sigma_{ij}(t)} = -\frac{w_j(t) \times v_i(t) \times \|x_i(t) - c_{ij}(t)\|^2}{\sigma_{ij}^2(t)}, \quad (26)$$

$$\frac{\partial e(t)}{\partial w_j(t)} = -v_j(t), \quad j = 1, 2, \cdots, p. \quad (27)$$

根据式(11) ~ (17) 可以计算出 FNN 参数  $c, \sigma, w$  更新量。

### 1.3 HG-FNN 具体流程

为了评价模糊神经网络性能, 定义 RMSE 作为评价网络性能指标

$$RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_d(t) - y(t))^2}. \quad (28)$$

根据以上对模糊神经网络的参数学习, 具体的 HG-FNN 算法流程如下。

Step 1: 初始化最大迭代次数  $\rho$ , 模糊神经网络隐含层节点数  $p$ , 学习率常数  $\alpha, \gamma$ . 随机初始化 FNN 隶属函数中心  $c$ 、宽度  $\sigma$ 、输出权值  $w$ .

Step 2: 当前样本进入模糊神经网络训练, 根据式(1) ~ (5) 计算模糊神经网络输出  $y(t)$ , 由式(6) 得到网络误差  $e(t)$ ; 根据式(23) ~ (27) 完成对 FNN 参数梯度的求解。

Step 3: 由式(18) ~ (20) 计算学习率  $\eta_1(t), \eta_2(t), \eta_3(t)$ , 根据式(15) ~ (17) 计算下一时刻 FNN 的参数  $c, \sigma, w$ , 并根据式(28) 计算 RMSE 值。

Step 4: 判断 RMSE 值是否满足停止条件或满足达到最大迭代次数, 若满足, 则停止计算, 否则转至 Step 2 进行重新训练。

混合梯度下降算法为研究 FNN 参数优化问题提供了一种新方法. 由式(8) ~ (12) 可知, 该算法通过泰勒级数展开的方法设计了一种范数自适应学习率, 并将其应用在 FNN 中  $c, \sigma, w$  参数对应的  $\eta_1(t), \eta_2(t), \eta_3(t)$  自适应学习率, 通过式(22) ~ (27) 完成对 FNN 参数梯度的求解. 式(15) ~ (17) 通过对各参数的调整从

而实现网络参数的自校正. 与一般BP算法中的自适应学习率不同: 1) 该自适应学习率通过泰勒级数推导出FNN参数调整的最优步长, 具有明显的数学意义; 2) 该自适应学习率通过FNN参数雅克比矩阵和网络误差两种信息结合而成, 雅克比矩阵和网络误差的变化成为下一时刻的最优步长; 3) 基于该自适应学习率的梯度算法可以保证李雅普诺夫函数值快速减小, 保证了FNN的收敛速度, 具体见收敛性分析.

## 2 收敛性分析

参数优化算法的收敛性关系到最终神经网络性能, 为了保证FNN的成功应用, 以下给出混合算法的收敛性证明. 定义网络误差

$$e(t) = y_d(t) - y(t). \quad (29)$$

**定理1** 如果任意初始FNN参数 $\theta(1)$ ,  $t$ 时刻更新为

$$\theta(t) = \theta(1) + \int_1^t \Delta\theta(t)dt, \quad (30)$$

$$\theta(t) = \frac{\|e(t)\|^2}{\|\mathbf{J}(t)e(t)\|^2} \mathbf{J}(t)e(t), \quad (31)$$

则能够保证HG-FNN收敛, 即

$$\lim_{t \rightarrow \infty} e(t) = 0. \quad (32)$$

其中:  $\theta$ 为FNN参数 $c$ 、 $\sigma$ 、 $w$ ,  $\Delta\theta$ 为FNN参数的更新量.

**证明** 定义李雅普诺夫(Lyapunov)函数

$$\Delta V(t) = \frac{1}{2}e^2(t). \quad (33)$$

由式(22)可知, 李雅普诺夫函数 $V(t)$ 的变化量为

$$\Delta V(t) = -e(t) \frac{\partial y(t)}{\partial \theta(t)} \Delta\theta(t) = -e(t) \mathbf{J}(t) \Delta\theta(t). \quad (34)$$

将式(31)代入(34), 得到

$$\Delta V(t) = -\|e(t)\|^2 \leq 0. \quad (35)$$

由式(34)可知, 当 $e(t) \neq 0$ 时,  $\Delta V(t) < 0$ . 如果 $\Delta V(t)$ 连续一致有界, 则根据Barbalat's引理可知,  $t \rightarrow \infty$ ,  $\Delta V(t) \rightarrow 0$ ,  $e(t) \rightarrow 0$ .  $\square$

上述讨论给出了算法的收敛性证明. 如果 $\Delta\theta(t)$ 非0, 则收敛性得以保证, 同时 $\Delta\theta(t)$ 非0意味着 $\|\mathbf{J}(t)e(t)\|$ 不为0. 如果 $\|\mathbf{J}(t)e(t)\| = 0$ , 则表明陷入了局部最小值. 因此, 定理1表明, 只要在FNN训练过程中避免局部最小, 则FNN有机会达到全局最小值. 式(35)显示, 李雅普诺夫能量函数值的改变量恒等于负的网络误差的范数平方, 表明能量函数值是不断减小的, 而且网络误差越大, 函数值下降得越迅速, 保证了FNN的快速收敛.

## 3 实验分析

为了验证HG-FNN的非线性建模能力, 将所设计的HG-FNN应用于混沌系统辨识、Mackey-Glass混沌时间序列预测的经典案例和污水处理过程关键水质参数预测, 并与其他方法在测试精度以及收敛速度等指标进行对比.

为了充分体现HG-FNN的性能, 利用平均百分比误差(APE)计算网络的精确度

$$\text{APE}(t) = \frac{1}{N} \sum_{t=1}^N \left\| \frac{e(t)}{y_d(t)} \right\| \times 100\%. \quad (36)$$

其中:  $e(t)$ 为 $t$ 时刻网络误差,  $y_d(t)$ 为 $t$ 时刻期望输出.

### 3.1 混沌系统辨识

Henon系统是具有一个时间延迟和两个参数的二阶模型, 混沌系统如下<sup>[19]</sup>:

$$y(t+1) = -H \cdot y^2(t) + Q \cdot y(t-1) + 1.0. \quad (37)$$

其中:  $H = 1.4$ ,  $Q = 0.3$ . 选取式(28)中2000组样本, 1000组作为训练样本, 1000组作为测试样本. FNN输入为 $y(t-1)$ 和 $y(t)$ , 输出为 $y(t+1)$ .

仿真结果如图1~图3所示, 图1为程序运行50次, 每运行一次网络收敛到期望误差时的迭代次数, 基于混合梯度下降算法的模糊神经网络(HG-FNN)和基于传统梯度下降算法的模糊神经网络(FG-FNN)的收敛速度对比. 图2为HG-FNN和FG-FNN测试样本的测试结果. 图3为HG-FNN和FG-FNN的预测误差. 本实验 $\alpha, \gamma$ 均为0.08.

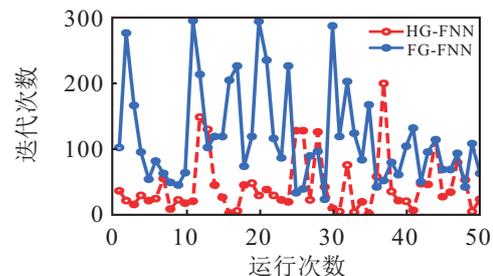


图1 收敛速度对比

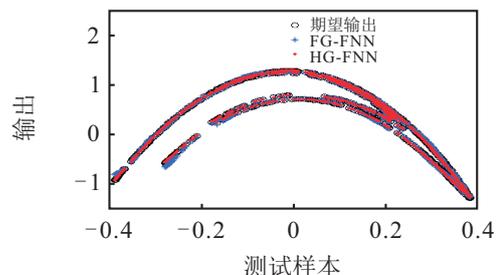


图2 测试结果

与基于3种自适应梯度下降算法和其他算法的FNN进行性能指标比较, 比较结果如表1所示.

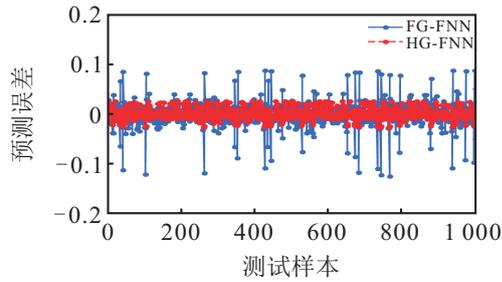


图3 预测误差

表1 不同算法性能对比

算法	训练RMSE	测试RMSE	测试APE	epoch
HG-FNN	0.0107	0.0073	0.0227	18
FG-FNN	0.0118	0.0161	0.0560	67
OG-FNN <sup>[8]</sup>	0.0155	0.0117	0.0261	51
DLR-FNN <sup>[11]</sup>	0.0134	0.0109	0.0257	79
ILR-FNN <sup>[9]</sup>	0.0117	0.0123	0.0550	40
RSEFNN-LF <sup>[19]</sup>	0.0320	0.0230	—	—
IRSFNN <sup>[20]</sup>	0.0160	0.0140	—	—

如表1所示,HG-FNN的训练RMSE为0.0107、测试RMSE为0.0073、测试APE为0.0227、同时达到期望误差0.02的平均迭代epoch为18. HG-FNN较FG-FNN在泛化能力和收敛速度性能上都有很大提高.与OG-FNN<sup>[8]</sup>、DLR-FNN<sup>[11]</sup>、ILR-FNN<sup>[9]</sup>3种自适应梯度下降算法相比,HG-FNN具有最小的训练RMSE、测试RMSE、测试APE,同时具有最快的收敛速度,具有明显的竞争优势.与基于最小二乘算法的递归模糊神经网络RSEFNN-LF<sup>[19]</sup>、递归模糊神经网络IRSFNN<sup>[20]</sup>比较,有极小的测试RMSE.表明了HG-FNN收敛速度较快,有较好的泛化能力和较强的非线性系统辨识能力.

### 3.2 Mackey-Glass 时间序列预测

Mackey-Glass混沌时间序列方程如下:

$$x(t+1) = (1-a)x(t) + \frac{bx(t-\tau)}{1+x^{10}(t-\tau)}. \quad (38)$$

其中: $a = 0.1, b = 0.2, \tau = 17$ . 初始状态 $x(0) = 1.2$ , 预测模型描述为

$$x(t+p) = f[x(t), x(t-\Delta), x(t-2\Delta), x(t-3\Delta)]. \quad (39)$$

其中: $p = 6, \Delta t = 6$ . 在 $t = 1$ 到 $t = 1000$ 之间选择500组样本作为训练样本,500组样本作为测试样本.

仿真结果如图4~图6所示.图4为HG-FNN和FG-FNN在Mackey-Glass时间序列预测收敛速度对比,图5为HG-FNN和FG-FNN训练过程中RMSE变化曲线.图6为HG-FNN和FG-FNN的预测误差.本实验 $\alpha = 0.07, \gamma = 0.1$ .

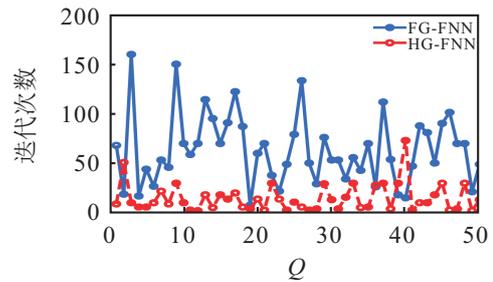


图4 收敛速度对比

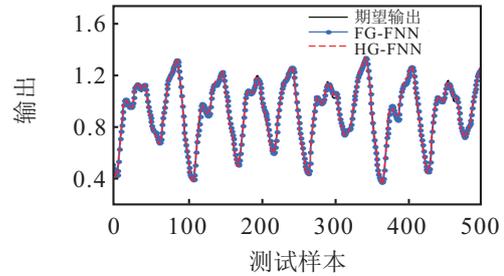


图5 测试结果

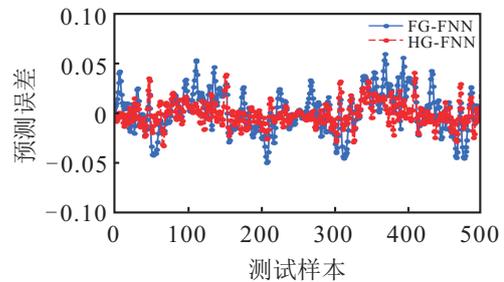


图6 预测误差

与基于3种自适应梯度下降算法和其他算法的FNN进行性能指标比较,比较结果如表2所示.

表2 不同算法性能对比

算法	训练RMSE	测试RMSE	测试APE	epoch
HG-FNN	0.0109	0.0085	0.0052	19
FG-FNN	0.0121	0.0148	0.0101	65
OG-FNN <sup>[8]</sup>	0.0154	0.0116	0.0076	39
DLR-FNN <sup>[11]</sup>	0.0218	0.0165	0.0098	107
ILR-FNN <sup>[9]</sup>	0.0098	0.0113	0.0074	36
GP-FNN <sup>[12]</sup>	0.0125	0.0107	0.0076	41
FAOSPFNN <sup>[14]</sup>	0.0073	0.0127	0.0090	—
SOFMLS <sup>[15]</sup>	—	0.0471	0.0335	—
SOFNNGA <sup>[16]</sup>	0.0144	0.0132	0.0094	—

由表2可见:HG-FNN的训练RMSE为0.0109、测试RMSE为0.0085、测试APE为0.0052、同时达到期望误差0.02的平均迭代epoch为19.尤其是测试RMSE、收敛速度方面,HG-FNN较FG-FNN有很大的提高;与OG-FNN<sup>[8]</sup>、DLR-FNN<sup>[11]</sup>、GP-FNN<sup>[12]</sup>等自适应梯度下降算法相比,具有明显的优势;与FAOSPFNN<sup>[14]</sup>、SOFNNGA<sup>[16]</sup>等非梯度下降算法比较,具有极小的测试RMSE.表明了HG-FNN具有较好的

泛化能力和较快的收敛速度,体现了HG-FNN预测Mackey-Glass混沌时间序列的优越性,与其他算法相比,具有明显的优势。

### 3.3 污水处理出水氨氮参数预测

污水处理出水氨氮含量是衡量水污染的重要指标.针对氨氮在水处理中的非线性、时变性和滞后性导致难以实时检测的问题<sup>[21-24]</sup>,提出一种基于HG-FNN的出水氨氮(NH<sub>4</sub>-N)的预测模型.通过机理分析,得出NH<sub>4</sub>-N的辅助变量,模型输入为温度 $T$ 、厌氧末端ORP、好氧前端DO、好氧末端TSS和出水PH,输出为NH<sub>4</sub>-N.

为了测试NH<sub>4</sub>-N预测模型的有效性,实验选取来自北京某污水处理厂的140组NH<sub>4</sub>-N样本数据,其中90组作为训练样本,50组作为测试样本.仿真结果如图7和图8所示.图7为HG-FNN和FG-FNN测试样本的测试结果,图8为HG-FNN和FG-FNN的预测误差.本实验 $\alpha = 0.02, \gamma = 0.02$ .

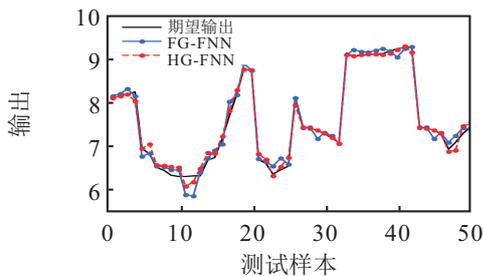


图7 测试结果

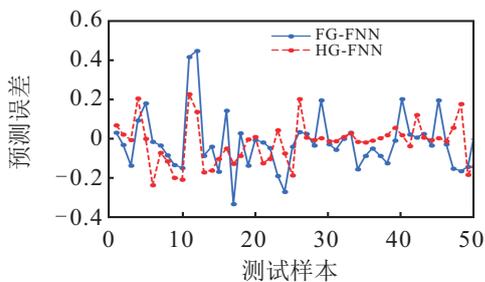


图8 预测结果

利用HG-FNN对NH<sub>4</sub>-N建模预测,与ANFIS等模型进行性能指标比较,结果如表3所示.准确度(accuracy)公式如下所示:

$$\text{accuracy} = \frac{1}{N} \sum_{t=1}^N 1 - \frac{|y_d(t) - y(t)|}{y_d(t)} \times 100\%. \quad (40)$$

由表3可见,HG-FNN预测NH<sub>4</sub>-N时的训练RMSE为0.0359、测试RMSE为0.1064,准确度高达98.80%,与OG-FNN<sup>[8]</sup>、DLR-FNN<sup>[11]</sup>、GP-FNN<sup>[12]</sup>等自适应梯度下降算法相比,具有最小的测试RMSE、最高的准确率,明显优于其他算法.表明了HG-FNN应用于NH<sub>4</sub>-N参数预测时,具有较高的预测精度和较快的收敛速度,验证了HG-FNN在预测污水处理中

出水氨氮参数的有效性和优越性。

表3 不同算法性能对比

算法	训练RMSE	测试RMSE	准确度
HG-FNN	0.0359	0.1064	98.80
FG-FNN	0.0510	0.1434	96.75
OG-FNN <sup>[8]</sup>	0.0922	0.3066	94.67
DLR-FNN <sup>[11]</sup>	0.0807	0.3823	91.23
ILR-FNN <sup>[9]</sup>	0.0465	0.1770	98.38
ANFIS <sup>[24]</sup>	—	1.210	97.20
GP-FNN <sup>[12]</sup>	0.0783	0.1919	96.30

## 4 结论

本文为了提高模糊神经网络收敛速度和泛化能力,提出了一种基于混合梯度下降算法的模糊神经网络(HG-FNN).HG-FNN在保证神经网络收敛性能的前提下实现参数在线调整,提高了神经网络的自适应能力.通过非线性系统建模与污水处理关键参数建模研究,以及与自适应梯度下降算法非梯度下降算法相比较,得到以下结论:

1)通过设计FNN参数调整过程的自适应学习率,利用链式法则完成了对FNN参数梯度的求解,实现了FNN的参数自校正.

2)与一般BP算法中的自适应学习策略不同,当误差 $e \neq 0$ 时,自适应学习率使李雅普诺夫能量函数的改变量恒等于负的网络误差的绝对值,使能量函数值不断减小,网络误差越大,李雅普诺夫能量函数值下降越迅速,保证了收敛速度.当 $t \rightarrow \infty$ 时,李雅普诺夫函数值达到最小,保证了其泛化能力.

3)文中提出的HG-FNN稳定高效,适用于非线性系统的建模预测,为复杂系统的建模预测提供了一种行之有效的方法.

## 参考文献(References)

- [1] Chen J, Sun F, Sun Y, et al. Modeling and controller design for complex flexible nonlinear systems via a fuzzy singularly perturbed approach[J]. Information Sciences, 2014, 255(10): 187-203.
- [2] Hou Z S, Wang Z. From model-based control to data-driven control: Survey, classification and perspective[J]. Information Sciences, 2013, 235(20): 3-35.
- [3] Wu J F, Li Y, Quevedo D E, et al. Data-driven power control for state estimation: A bayesian inference approach[J]. Automatica, 2015, 54 (1): 332-339.
- [4] 夏国清, 杨莹, 赵为光. 欠驱动AUV模糊神经网络 $L_2$ 增益鲁棒跟踪控制[J]. 控制与决策, 2013, 28(3): 351-356.

- (Xia G Q, Yang Y, Zhao W G.  $L_2$  gain robust tracking control for under actuated AUV fuzzy neural network[J]. Control and Decision, 2013, 28(3): 351-356.)
- [5] Kayacan E, Kayacan E, Khanesar M A. Identification of nonlinear dynamic systems using type-2 fuzzy neural networks-A novel learning algorithm and a comparative study[J]. IEEE Trans on Industrial Electronics, 2015, 62(3): 1716-1724.
- [6] 乔俊飞, 袁喜春, 韩红桂. 基于EKF的自组织T-S模糊Elman网络[J]. 控制与决策, 2014, 29(5): 853-859. (Qiao J F, Yuan X C, Han H G. Self-organizing T-S fuzzy Elman network based on EKF[J]. Control and Decision, 2014, 29(5): 853-859.)
- [7] Han H G, Li Y, Qiao J F. A fuzzy neural network approach for online fault detection in waste water treatment process[J]. Computers and Electrical Engineering, 2014, 40(7): 2216-2226.
- [8] Wu W, Feng G, Li Z, et al. Deterministic convergence of an online gradient method for BP neural networks[J]. IEEE Trans on Neural Networks, 2005, 16(3): 533-540.
- [9] Zhang X, Wu L. Handwritten digit recognition based on improved learning rate BP algorithm[C]. Shanghai: IEEE, 2010: 1-4.
- [10] Meng D, Pei Z. Dynamic adaptive learning algorithm based on two-fuzzy neural-networks[J]. Neurocomputing, 2014, 125(1): 88-94.
- [11] Zhang R, Xu Z B, Huang G B. Global convergence of online BP training with dynamic learning rate[J]. IEEE Trans on Neural Networks and Learning Systems, 2012, 23(2): 330-341.
- [12] Han H G, Qiao J F. A self-organizing fuzzy neural network based on a growing-and-pruning algorithm[J]. IEEE Trans on Fuzzy Systems, 2010, 18(6): 1129-1143.
- [13] Davanipoor M, Zekri M, Sheikholeslam F. Fuzzy wavelet neural network with an accelerated hybrid learning algorithm[J]. IEEE Trans on Fuzzy Systems, 2012, 20(3): 463-470.
- [14] Wang N, Er M J, Meng X Y. A fast and accurate online self-organizing scheme for parsimonious fuzzy neural networks[J]. Neurocomputing, 2009, 72(16/17/18): 3818-3829.
- [15] Rubio J de Jesús. SOFMLS: Online self-organizing fuzzy modified least-squares network[J]. IEEE Trans on Fuzzy Systems, 2009, 17(6): 1296-1309.
- [16] Leng G, McGinnity T M, Prasad G. Design for self-organizing fuzzy neural networks based on genetic algorithms[J]. IEEE Trans on Fuzzy Systems, 2006, 14(6): 755-761.
- [17] Mashinchi M R, Selamat A. An improvement on genetic-based learning method for fuzzy artificial neural networks[J]. Applied Soft Computing, 2009, 9(4): 1208-1216.
- [18] Kuo R J, Hung S Y, Cheng W C. Application of an optimization artificial immune network and particle swarm optimization-based fuzzy neural network to an RFID-based positioning system[J]. Information Sciences, 2014, 262(20): 78-98.
- [19] Juang C F, Lin Y Y, Tu C C. A recurrent self-evolving fuzzy neural network with local feedbacks and its application to dynamic system processing[J]. Fuzzy Sets and Systems, 2010, 161(19): 2552-2568.
- [20] Lin Y Y, Chang J Y, Lin C T. Identification and prediction of dynamic systems using an interactively recurrent self-evolving fuzzy neural network[J]. IEEE Trans on Neural Networks and Learning Systems, 2013, 24(2): 310-321.
- [21] Dürrenmatt D J, Gujer W. Data-driven modeling approaches to support wastewater treatment plant operation[J]. Environmental Modelling & Software, 2012, 30: 47-56.
- [22] Huang M, Wan J, Ma Y, et al. A fast predicting neural fuzzy model for on-line estimation of nutrient dynamics in an anoxic/oxic process[J]. Bioresource Technology, 2010, 101(6): 1642-1651.
- [23] 乔俊飞, 韩红桂. RBF神经网络的结构动态优化设计[J]. 自动化学报, 2010, 36(6): 865-872. (Qiao J F, Han H G. Optimal structure design for RBFNN structure[J]. Acta Automatica Sinica, 2010, 36(6): 865-872.)
- [24] 胡康, 万金泉, 马邕文, 等. 基于模糊神经网络的A2/O工艺出水氨氮在线预测模型[J]. 中国环境科学, 2012, 32(2): 260-267. (Hu K, Wan J Q, Ma Y W, et al. Online prediction model based on fuzzy neural network for the effluent ammonia concentration of A2/O system[J]. China Environmental Science, 2012, 32(2): 260-267.)

(责任编辑: 郑晓蕾)