

# 基于模型输出敏感度分析的动态神经网络结构设计

韩红桂, 李 森, 乔俊飞

(北京工业大学电子信息与控制工程学院, 北京 100124)

**摘 要:** 神经网络的性能由其训练算法和拓扑结构共同确定. 为了解决设计网络结构的动态调整问题, 论文给出了一种神经网络结构动态设计方法. 以隐含层神经元输出的贡献对模型输出敏感度进行分析, 从而调整神经网络结构, 对贡献太小的神经元予以删除, 对贡献值太大的神经元利用最邻近法在其附近插入新的神经元. 通过对非线性函数进行逼近和对非线性系统关键参数进行预测证明了该方法的有效性.

**关键词:** 神经网络; 动态结构设计; 模型输出敏感度分析

**中图分类号:** TP183      **文献标识码:** A      **文章编号:** 0372-2112 (2010) 03-0731-06

## Design of Dynamic Neural Network Based on the Sensitivity Analysis of Model Output

HAN Hong-gui, LI Miao, QIAO Jun-fei

(College of Electronic and Control Engineering, Beijing University of Technology, Beijing 100124, China)

**Abstract:** The capabilities of neural networks are influenced by the learning algorithms and the topologies. Thus, in order to solve the problem of dynamic topologies, a new design method for dynamic structure of neural network is proposed in this paper. The dynamic design for neural network is based on the sensitivity analysis (SA) of the model output. This algorithm can delete the nodes in the hidden layer whose contribution ratios are too little; and add new nodes to the hidden layer whose ratios are too large relied on the the nearest neighbor interpolation. Finally, This proposed algorithm is used to track the nonlinear functions and predict the nonlinear systems, the results demonstrate the good effect of the dynamic feed-forward neural network (SAFNN).

**Key words:** neural network; design of dynamic structure; sensitivity analysis of model output

### 1 引言

神经网络由于具有较好的学习能力, 在非线性系统的控制、建模、辨识等领域得到广泛的应用<sup>[1-3]</sup>. 神经网络的性能主要取决于神经网络的结构与其参数学习算法; 由于神经网络的结构在研究过程中一般都认为不变, 使得神经网络的总体性能受到很大的限制, 这已经成为神经网络研究中的一个关键问题. 文献[4]指出规模过大的神经网络(例如: BP, RBF, Hopfield等)具有较好的学习精度, 但是往往会出现过拟合现象, 而且需要较大的存储空间和计算时间; 而文献[5]则给出规模过小的神经网络虽具有较简单的网络结构, 但处理复杂问题的能力又显得不足.

针对以上问题, 动态神经网络设计已经成为神经网络研究中的一个热点, 动态神经网络主要由神经网络修剪与生长两部分组成<sup>[6]</sup>. 修剪型神经网络, 通过在线修改神经网络的参数, 减去神经网络中冗余神经元的策

略, 提高网络性能, 解决由于神经元过多而过拟合的问题. 生长型神经网络是通过自动增加神经网络中神经元个数或改变神经网络的拓扑结构的策略, 改进神经网络性能, 提高神经网络的信息处理能力. 文献[7]提出一种修剪型神经网络算法, 其删减原理是当训练完所有的样本后达到一定的误差精度时对神经网络隐含层神经元进行删减, 删减是训练结束时进行而不是在学习过程中进行的. 因此, 该动态神经网络设计方法基本不适合实时性较强的系统. 文献[8]提出了一种增长型神经网络模型, 该方法能够通过训练底层网络的方法对神经网络隐含层的神经元进行改变. 然而该方法受初始值的影响较大, 有时最终神经网络的性能不稳定, 经常会遗漏最佳神经网络, 陷入局部最优; 另外, 文献[9, 10]能够对神经网络中隐含层神经元的个数进行修剪和生长, 取得了不错的效果, 但是神经网络结构修改机制的确定需要一定的经验.

为此, 本文将引入一种基于敏感度分析的动态前

馈神经网络 (Feed-forward Neural Network based on Sensitivity Analysis, SAFNN) 结构设计方法,通过对神经网络输出敏感度分析,删除敏感度小的隐含层神经元,同时利用最近邻插值法对敏感度大的神经元进行拆分.最后结合最速下降算法对神经网络连接权值进行修改,获得了较精简的结构和较好的性能,利用其对非线性函数逼近和非线性系统软测量和预测都取得了较好的结果.

## 2 前馈神经网络

前馈神经网络结构一般由输入层、隐含层、输出层组成,其中隐含层一般有一层或多层组成,本文只讨论含有一层隐含层的情况,其结构如图 1 所示(多输入单输出).神经网络各层的具体功能如下:

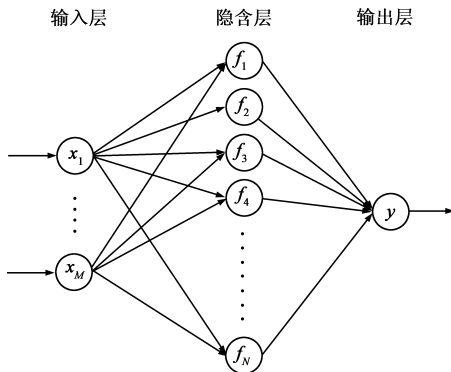


图1 前馈神经网络结构图

第一层:输入层

该层有  $M$  个节点,分别是输入  $x_1, x_2, \dots, x_M$ .

$$In_i^{(1)}(k) = x_i(k), Out_i^{(1)}(k) = In_i^{(1)}(k) \quad (1)$$

其中,  $i = 1, 2, \dots, M$ ,  $In_i^{(1)}(k)$ ,  $Out_i^{(1)}(k)$  分别表示  $k$  时刻输入层第  $i$  个神经元的输入和输出.

第二层:隐含层

该层对输入量进行处理,有  $N$  个神经元.

$$In_i^{(2)}(k) = \sum_{i=1}^M w_{j,i}^2(k) x_i(k), Out_j^{(2)}(k) = f_j(In_i^{(2)}(k))$$

$$(i = 1, 2, \dots, M; j = 1, 2, \dots, N) \quad (2)$$

其中函数  $f(x) = \frac{1}{1 + e^{-x}}$ ,  $In_j^{(2)}(k)$ ,  $Out_j^{(2)}(k)$  分别表示  $k$  时刻隐含层第  $j$  个神经元的输入和输出,  $w_{j,i}^2$  为输入层第  $i$  个神经元与隐含层第  $j$  个神经元间的联结权值.

第三层:输出层

该层为了描述方便,只设有一个输出神经元,其输出可以由下式来计算:

$$y(k) = Out^{(3)}(k) = \sum_{j=1}^N w_j^3(k) Out_j^{(2)}(k). \quad (3)$$

其中,  $j = 1, 2, \dots, N$ ,  $w_j^3$  表示隐含层第  $j$  个神经元和输出层神经元间的联结权值.

## 3 动态 FNN 设计

动态 FNN 神经网络设计算法主要从两个方面考虑:神经元的生长和修剪;主要包括两个机制:神经网络中隐含层神经元的输出权值的敏感度分析;基于敏感度的隐含层神经元的增减.

敏感度分析 (Sensitivity Analysis, SA)<sup>[11]</sup> 主要根据模型或系统的输出分析出模型或系统中参数尤其是输入对输出的价值,从而根据期望值对其进行修改,直到达到期望值.敏感度分析 (SA) 虽然有不同的分析方法<sup>[12,13]</sup>,但是基本思想类似.该方法比较适合于模型不清楚或系统动力学不确定的问题.神经网络内部结构作为一种动力学问题,可以利用敏感度分析法计算其内部参数对神经网络输出的影响.

### 3.1 隐含层神经元的敏感度分析

本文提出的动态神经网络设计是利用敏感度分析 (SA) 的特点分析神经网络的隐含层和输出层间的联结权值对神经网络的输出的影响,剔除冗余的权值,从而删除与其相连接的隐含层神经元;同时通过最邻近插值法对作用太大的联结权值进行拆分,从而增加隐含层的神经元,以达到简化神经网络结构和提高神经网络性能的目的.神经网络可以分为两个部分,第一部分是输入层与隐含层连接,第二部分是隐含层与输出层连接,分解结构如图 2 所示.本文将重点介绍第二部分的运行机制.

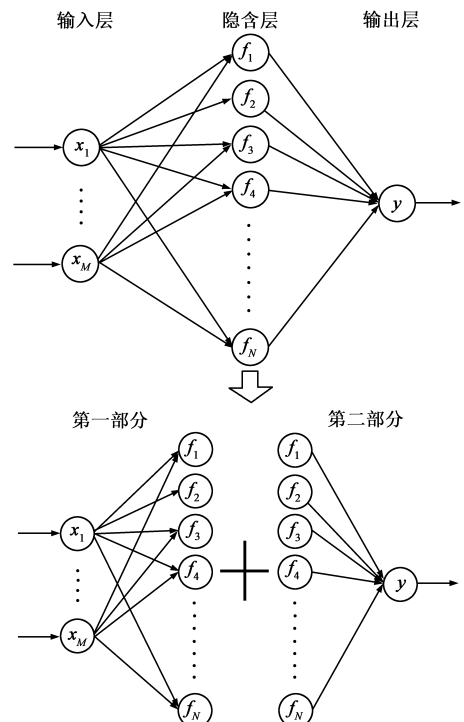


图2 前馈神经网络结构分解图

隐含层神经元输出权值  $w_h^3 (h = 1, 2, \dots, N)$  作为敏感度法的输入量, 利用式(4)计算权值对输出所做的贡献:

$$S_h = \frac{\text{Var}_{w_h^3} [E(y | \mathbf{w}^3 = w_h^3)]}{\text{Var}(y)} \quad (4)$$

其中,  $\mathbf{w}^3$  是输入向量,  $y$  是输出量,  $E(y | \mathbf{w}^3 = w_h^3)$  是  $\mathbf{w}^3$  等于  $w_h^3$  时  $y$  的值,  $y = F(w_1^3, w_2^3, \dots, w_N^3)$ ,  $\text{Var}_{w_h^3}$  是  $w_h^3$  的方差,  $S_h$  是  $w_h^3$  对应的权值对输出所做的贡献量. 对输入量  $w_h^3$  进行傅里叶变换(其中  $w_h^3$  的范围是  $[a_h, b_h]$ ):

$$w_h^3 = \frac{a_h + b_h}{2} + \frac{b_h - a_h}{\pi} \arcsin(\sin(\omega_h s)) \quad (5)$$

其中,  $\omega_h$  是选择的合适的频率, 通过变换, 输出可以变换为:

$$y = F(s) = \sum_{j=1}^{+\infty} (A_j \cos(\omega_j s) + B_j \sin(\omega_j s)) \quad (6)$$

其中,  $-\pi < s < \pi$ ,  $A_j = \frac{1}{2\pi} \int_{-\pi}^{\pi} F(s) \cos(\omega_j s) ds$ ,

$$B_j = \frac{1}{2\pi} \int_{-\pi}^{\pi} F(s) \sin(\omega_j s) ds.$$

从而, 公式(4)可变换为:

$$\text{Var}(y) = 2 \sum_{k=1}^{+\infty} (A_k^2 + B_k^2),$$

$$\text{Var}_{w_h^3} [E(y | \mathbf{w}^3 = w_h^3)] = 2 \sum_{k=1}^{+\infty} (A_{k\omega_h}^2 + B_{k\omega_h}^2),$$

$$S_h = \frac{\sum_{k=1}^{+\infty} (A_{k\omega_h}^2 + B_{k\omega_h}^2)}{\sum_{k=1}^{+\infty} (A_k^2 + B_k^2)} \quad (7)$$

另外, 为了减少偏差对结果的判断产生的误差, 利用归一化对  $S_h$  进行处理:

$$ST_h = \frac{S_h}{\sum_{h=1}^N S_h} \quad (8)$$

通过以上计算可以得到每个隐含层神经元输出权值对神经网络输出的敏感度, 该敏感度可以作为神经网络结构动态设计的依据.

### 3.2 神经网络生长和修剪

根据权值敏感度分析结果, 神经网络的结构可以实现在线生长和修剪. 其主要思想为: 当神经网络结构存在冗余时将会修剪其冗余神经元, 当神经网络处理信息能力不够或权值太大时隐含层神经元将会生长. 同时, 利用最速下降算法对其参数进行修改. 因此, 初始神经网络隐含层的神经元数将对整个神经网络最终性能影响不是很大, 而且由于神经网络结构可以在线修改, 其性能较之静态神经网络有很大的提高. 神经网络的生长和修剪过程如下:

首先, 当  $ST_h \geq \epsilon_1$  时 ( $\epsilon_1$  的值预先给出), 表明隐含层的第  $h$  个神经元输出权值太大, 对其进行拆分; 同时, 为保证神经网络的收敛速度, 新增加的神经元要求相对均匀, 因此, 对输出权值太大的神经元进行拆分:

$$w_{\text{new}}^3 = \frac{1}{2} (w_h^3 + w_{h \cdot \text{nearest}}^3),$$

$$w_{\text{new}i}^2 = \frac{1}{2} (w_{hi}^2 + w_{h \cdot \text{nearest}i}^2), (i = 1, 2, \dots, M) \quad (9)$$

其中  $w_{\text{new}}^3$ ,  $w_{\text{new}i}^2$  为新插入神经元的权值,  $w_{h \cdot \text{nearest}}^3$ ,  $w_{h \cdot \text{nearest}i}^2$  为与神经元  $h$  敏感度最接近的神经元的权值.

其次, 当  $ST_h \leq \epsilon_2$  时 ( $\epsilon_2$  的值预先给出), 表明隐含层的第  $h$  个神经元输出权值对整个网络输出的影响不足, 可认为该神经元为冗余神经元, 删除该神经元, 同时删除与其相连的所有权值.

最后, 当神经网络结构确定以后, 将会修改神经网络参数. 由于经过生长和修剪, 神经网络的结构相对较优, 为了提高整体神经网络训练速度, 甚至应用于在线学习, 利用最速下降算法对神经网络的参数进行修改.

基于输出敏感度分析的动态前馈神经网络结构设计算法的具体步骤如下:

**步骤 1** 给定一个隐含层神经元不为零的三层前馈神经网络, 进行训练; 判断是否满足结构修改步数, 是, 转向步骤 2, 否则转向步骤 6.

**步骤 2** 找出每一个隐含层神经元输出权值  $w_h^3$  的最大值和最小值;

**步骤 3** 利用公式(8)对每一个神经元输出权值  $w_h^3$  进行敏感度分析, 计算其对输出的贡献值;

**步骤 4** 删除贡献值小于  $\epsilon_2$  的隐含层神经元, 调整神经网络结构;

**步骤 5** 利用公式(9)对贡献值大于  $\epsilon_1$  的隐含层神经元进行拆分, 调整神经网络结构;

**步骤 6** 利用最速下降算法对神经网络的连接权值进行修改;

**步骤 7** 满足所有停止条件或达到计算步骤时停止计算, 否则转向步骤 6(不满足结构修改步数)或步骤 2(满足结构修改步数)进行重新训练.

值得注意的是该算法中结构的确定并不是每次循环运行都进行判断的, 而是每运行一定的步骤以后才判断一次或整个运行过程中只对神经网络结构调整一次, 具体步骤由研究对象的特性决定, 以减少运行时间. 该算法通过敏感度法, 把时域的问题转换到频域进行研究, 提供了一种研究神经网络结构问题的新方法, 较之一般的基于空间搜索或信息矩阵处理的算法在计算时间和存储空间上有一定的优势.

## 4 仿真实验

动态神经网络算法能够在线修改隐含层神经元的

个数,简化神经网络结构,得到与研究对象相适应的网络结构,提高了神经网络整体的性能.利用本文提出的动态前馈神经网络算法对非线性函数进行逼近和对非线性系统参数软测量和预测,以证明该算法的有效性;实验结果与综合增长删减神经网络(IGPFNN)<sup>[9]</sup>算法和自适应增长删减神经网络(AGPNN)<sup>[14]</sup>算法进行比较,以证明该算法的优越性.

#### 4.1 非线性逼近能力分析

为了证明本文提出的基于输出敏感度法的动态前馈神经网络具有较好的性能,非线性函数的选取比较

关键,这里选取以下函数:

$$z = 0.5 \times \sin(6x) \times e^{-y} \quad (10)$$

其中,  $-1 < x < 1$ ,  $-1 < y < 1$ ,该函数已经被证明非常适合用来检测神经网络的性能<sup>[15]</sup>.选取 200 组样本,100 组用来训练,另外 100 组用来检验.初始神经网络的隐含层神经元数是 2,20,50,初始联结权值为任意值.在此条件下进行训练,每训练 50 步进行网络结构修改,训练过程(误差变化过程)如图 3 所示,神经网络逼近结果如图 4 所示,训练后的神经网络隐含层剩余神经元如图 5 所示,与 IGPNN 和 AGPNN 的性能比较如表 1 所示.

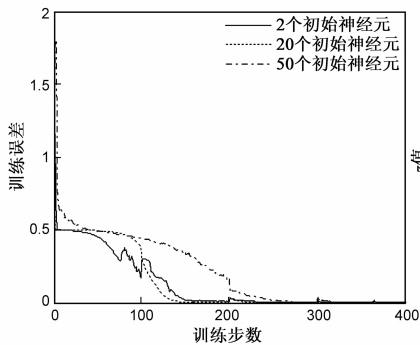


图3 三种不同初始结构神经网络训练过程

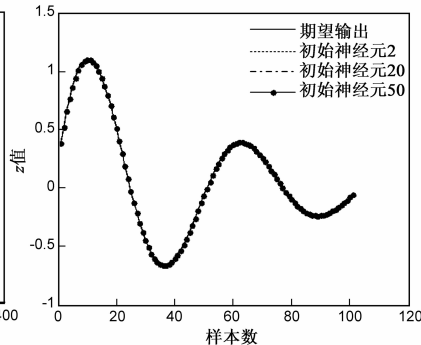


图4 三种不同初始结构神经网络逼近效果

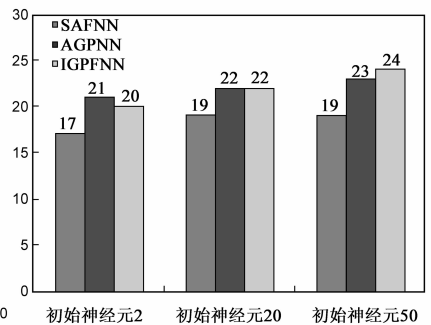


图5 三种不同初始结构神经网络剩余神经元

由图 3 可以发现基于敏感度分析的前馈神经网络(SAFNN)能够较快的达到期望误差,在训练过程中,随着神经网络中神经元的变化,神经网络在结构变化结束后的收敛速度较快.当神经网络的结构不断改变,最终达到合适的神经网络结构,神经网络的误差也不断收敛到期望值,图 3 详细描述了三种初始状态下误差变化过程,在初始隐含层神经元为 2 时需要 350 多步(约 2.31 分),在初始隐含层神经元为 20 时需要 370 多步(约 2.41 分),在初始隐含层神经元为 50 时需要 400 多步(约 2.56 分),从另一个侧面反映了在静态神经网络中初始结构对神经网络的最终性能有较大的影响.训练后的神经网络对该非线性函数的逼近效果如图 4 所示,该神经网络能够很好的逼近这个非线性函数,逼近误差为 0.011,从而也能看出网络输出值与函数值基本重合,具有很高的逼近能力.图 5 给出了三种算法在不同初始状态下最终神经网络中神经网络所剩余的神经元,图中详细描述了 SAFNN 中剩余神经元与初始神经网络结构关系不大.为了体现本文提出的动态前馈神经网络较之其他动态前馈神经网络有更好的性能,利用 SAFNN 与综合增长删减神经网络(IGPFNN)算法和自适应增长删减神经网络(AGPNN)算法进行比较,其详细比较结果如表 1,在相同的初始条件下,在达到相同的期望误差时 IGPNN 和 AGPNN 所需的训练时间比 SAFNN 多;而且训练后的神经网络较之 SAFNN 复杂,另外,存储空间也就相应增加;在利用训练后的神经网络

进行函数逼近时,IGPFNN 和 AGPNN 的检验误差也比 SAFNN 的大;因此,SAFNN 神经网络不但具有简单的网络结构,而且具有较强的非线性函数逼近能力.

表 1 三种算法性能比较

	初始网络 (隐含层)	期望误差	检验误差	最终网络 (隐含层)	训练时间 (分)
SAFNN	2	0.01	0.011	17	2.31
IGPFNN		0.01	0.012	21	4.55
AGPNN		0.01	0.013	20	6.23
SAFNN	20	0.01	0.011	19	2.41
IGPFNN		0.01	0.013	22	5.45
AGPNN		0.01	0.013	22	9.56
SAFNN	50	0.01	0.011	19	2.56
IGPFNN		0.01	0.012	23	6.22
AGPNN		0.01	0.011	24	10.89

#### 4.2 非线性系统软测量与预测

污水处理过程由于进水流量、水质成份及污染浓度波动剧烈,系统总是运行在非平稳状态,使得水质关键参数在线检测困难.本文利用 SAFNN 对污水处理过程中化学需氧量(COD)进行软测量与预测,COD 反映污水中有机污染的程度,可以提供有机底物、微生物和利用的溶解氧中的电子等价物之间的关联,是污水处理过程中一个非常关键的参数,也是污水处理过程中直接控制的参数,能否对 COD 进行实时监测已成为提高治污质量的关键.但是,由于测量手段的欠缺,目前污水处理厂不能及时反映污水处理实际情况,不能实

现对  $COD$  实时测量,从而限制了污水处理闭环控制系统的投用.而现存的传感器不但造价高、仪器寿命短,而且测量范围窄、稳定性差,因而  $COD$  的实时监测几乎成为污水水质监测的难点.

依据国际水质学会给出的活性污泥法污水处理模型 ASM1 模型,进水中总  $COD$  的组成如下:

$$COD = S_s + X_s + X_I + S_I \quad (11)$$

其中:  $S_s$  为易生物降解基质;  $X_s$  为慢速可生物降解基质;  $X_I$  为颗粒性惰性有机物质;  $S_I$  为可溶性惰性有机物质.根据生化反应过程机理,采用主元分析法确定对  $COD$  参数影响大且易于测量的因素作为辅助变量,本文选择固体悬浮物、PH 值、油、氨氮作为神经网络的输入变量,  $COD$  作为神经网络的输出变量.然后,对辅助变量进行数据采集,并将采集得到的数据进行适当地预处理.建立初始网络结构为 4-20-1 的前馈神经网络,神经网络每运行 200 步进行结构调整.为了检验对  $COD$  数据测量的准确性,本文采用某小型污水处理厂

2006 年的运行数据进行仿真,通过预处理选取其中 300 组作为实验数据,100 组用作测量,100 组用作预测,实际水温在  $20^\circ\text{C}$  左右.训练过程如图 6 所示(误差变化),训练后的神经元及其权值如图 7 所示,图 8 给出  $COD$  的软测量结果及其误差,  $COD$  的预测结果及其误差如图 9.

仿真结果表明:图 6 详细地描述了达到期望误差需要 4000 步(约 14.4 分)左右.图 7 给出了最终神经网络中隐含层的神经元为 21 个,这 21 个神经元是通过删除冗余的增加新神经元完成的,而且隐含层中神经元最多时为 35 个;图中详细描述了剩余神经元与输出神经元间的权值,这些权值比较均匀,从神经网络的性能方面看这是非常有利的;图 8 显示实测  $COD$  值与软测量结果基本吻合;另外,图 9 显示预测结果误差也控制在 5% 范围内,说明 SAFNN 能够有效的对污水处理过程中关键参数  $COD$  进行软测量和预测,为非线性系统中参数测量和预测提供了一种新的方法.

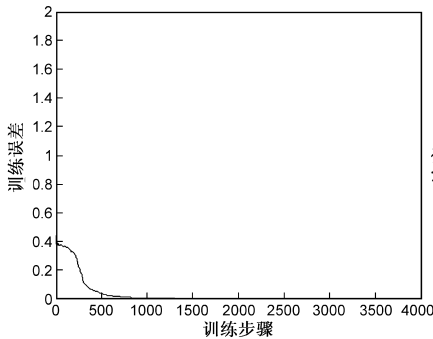


图6 神经网络训练过程

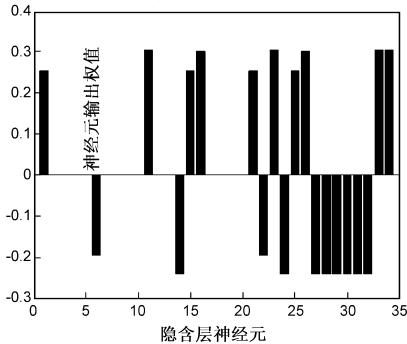
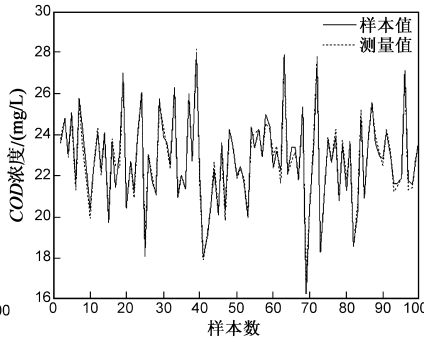
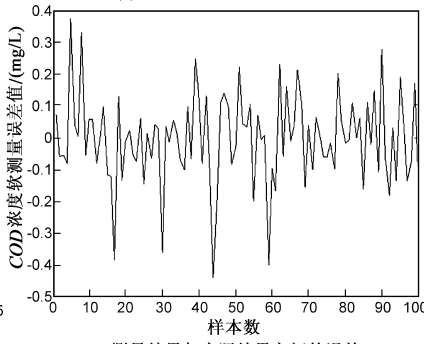


图7 最终神经网络隐含层神经元及其权值

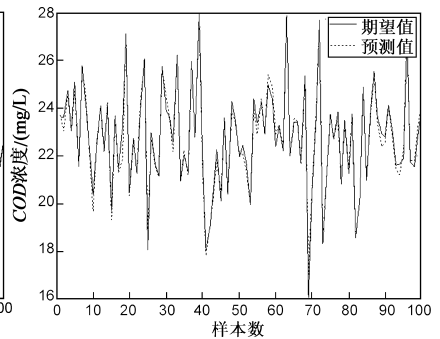


(a) 测量结果与实际结果的比较

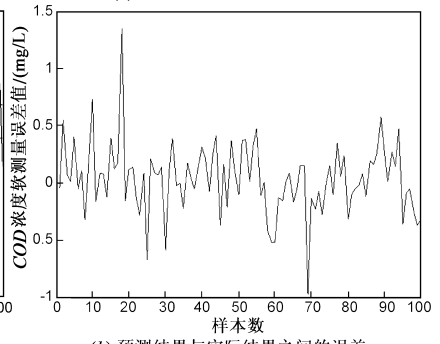


(b) 测量结果与实际结果之间的误差

图8  $COD$ 软测量结果



(a) 预测结果与实际结果的比较



(b) 预测结果与实际结果之间的误差

图9  $COD$ 预测结果

## 5 结论

针对一般神经网络在学习过程中网络结构不能改变的问题,本文提出了一种基于敏感度分析的动态前馈神经网络(SAFNN),利用其逼近非线性函数和对污水处理过程中关键参数  $COD$  进行预测,并与其他动态神经网络进行比较,详细分析了该算法的性能,得到以下结论:

(1)神经网络结构能够根据实际处理对象进行改

变,获得适合对象的动态神经网络算法,神经网络在其应用中对初始结构的依赖性得到解决;

(2)借鉴敏感度分析对隐含层神经元进行增减,获得的最终神经网络不但简洁,其性能也较好;提出了一种新的神经网络结构动态调整方式;

(3)利用该动态神经网络对污水处理过程中关键参数  $COD$  进行软测量和预测,获得了较好的效果,为非线性系统中关键参数测量和预测提供了一种新的方法.

## 参考文献:

- [1] 王小华, 何怡刚. 基于神经网络的 FRM 滤波器优化设计研究[J]. 电子学报. 2008, 36(3): 486 - 489.  
Wang Xiaohua, He Yigang. Optimal design of frequency-response-masking filters using neural networks[J]. Acta Electronica Sinica. 2008, 36(3): 486 - 489. (in Chinese)
- [2] S S Ge, C C Hang, T H Lee, T Zhang. Stable Adaptive Neural Network Control[M]. Boston, MA: Kluwer, 2001.
- [3] 田广, 戚飞虎. 移动摄像机环境下基于特征变换和 SVM 的分级行人检测算法[J]. 电子学报. 2008, 36(5): 1024 - 1028.  
TIAN Guang, QI Feihu. Feature transformation and SVM based hierarchical pedestrian detection with a monocular moving camera [J]. Acta Electronica Sinica. 2008, 36(5): 1024 - 1028. (in Chinese)
- [4] Di Wang. Fast constructive-covering algorithm for neural networks and its implement in classification [J]. Applied Soft Computing, 2008, 8(1): 166 - 173.
- [5] L Ma, K Khorasani. New training strategies for constructive neural networks with application to regression problems[J]. Neural Networks, 2004, 17(4): 589 - 609.
- [6] José Manuel Cano-Izquierdo, Julio Ibarrola, Miguel Pinzolas, Miguel Almonacid. DNSP: A biologically inspired dynamic Neural network approach to signal processing [J]. Neural Networks, 2008, 21(7): 1006 - 1019.
- [7] Xiaoqin Zeng, Daniel S Yeung. Hidden neuron pruning of multilayer perceptrons using a quantified sensitivity measure[J]. Neurocomputing, 2006, 69(7 - 9): 825 - 837.
- [8] Liying Ma, K Khorasani. Constructive feedforward neural networks using hermite polynomial activation functions[J]. IEEE Transaction on Neural Networks, 2005, 16(4): 821 - 833.
- [9] P L Narasimha, W H Delashmit, Michael T Manry, Jiang Li, F Maldonado. An integrated growing-pruning method for feedforward network training [J]. Neurocomputing, 2008, 71(13 - 15): 2831 - 2847.
- [10] Guang-Bin Huang, P Saratchandran, Narasimhan Sundararajan. A generalized growing and pruning RBF (GGAP-RBF) neural network for function approximation[J]. IEEE Transaction on Neural Networks, 2005, 16(1): 57 - 67.
- [11] Andrea Saltelli, Marco Ratto, Stefano Tarantola, Francesca Campolongo. Sensitivity analysis practices: Strategies for model-based inference[J]. Reliability Engineering and System Safety, 2006, 91(10 - 11): 1109 - 1125.
- [12] M Ratto, S Tarantola, A Saltelli. Sensitivity analysis in model calibration: GSA-GLUE approach[J]. Computer Physics Communications, 2001, 136(3): 212 - 224.
- [13] J Cariboni, D Gatelli, R Liska, A Saltelli. The role of sensitivity analysis in ecological modeling[J]. Ecological Modelling, 2007, 203(1 - 2): 167 - 182.
- [14] Chun-Fei Hsu. Adaptive growing-and-pruning neural network control for a linear piezoelectric ceramic motor[J]. Engineering Applications of Artificial Intelligence, 2008, 21(8): 1153 - 1163.
- [15] A Esposito, M Marinaro, D Oricchio, S Scarpetta. Approximation of continuous and discontinuous mappings by a growing neural RBF-based algorithm[J]. Neural Networks, 2005, 13(6): 651 - 665.

## 作者简介:



韩红桂 男, 2005 年于中国民航大学获得学士学位, 现为北京工业大学博士研究生, 主要研究方向为复杂过程建模与控制, 神经网络分析与设计等.

E-mail: rechar112@emails.bjut.edu.cn



乔俊飞 男, 1998 年于东北大学获博士学位, 现为北京工业大学教授, 博士生导师, 主要研究方向为智能控制理论及应用, 神经网络分析与设计等.

E-mail: isibox@sina.com