

文章编号: 1001-0920(2012)08-1265-04

## 基于混合进化算法的RBF神经网络时间序列预测

龙文<sup>1,2</sup>, 梁昔明<sup>2</sup>, 龙祖强<sup>3</sup>, 秦浩宇<sup>2</sup>

(1. 贵州财经学院 贵州省经济系统仿真重点实验室, 贵阳 550004; 2. 中南大学 信息科学与工程学院, 长沙 410083; 3. 衡阳师范学院 物理与电子信息科学系, 湖南 衡阳 421008)

**摘要:** 提出一种基于梯度下降法的混合进化算法, 用于确定径向基函数(RBF)神经网络结构和优化其参数. 在进化算法中嵌入梯度下降算子, 对每一代中若干个精英个体以一定概率利用梯度下降法进行搜索, 以加强算法的局部搜索能力. 利用混合进化算法对RBF网络结构和参数同时进行训练和优化, 对网络节点数和参数进行混合编码. 仿真实验结果表明该RBF网络具有较强的泛化能力.

**关键词:** 径向基函数神经网络; 进化算法; 梯度下降法; 非线性时间序列预测

中图分类号: TP273

文献标识码: A

## RBF neural network time series forecasting based on hybrid evolutionary algorithm

LONG Wen<sup>1,2</sup>, LIANG Xi-ming<sup>2</sup>, LONG Zu-qiang<sup>3</sup>, QIN Hao-yu<sup>2</sup>

(1. Guizhou Key Laboratory of Economics System Simulation, Guizhou College of Finance and Economics, Guiyang 550004, China; 2. School of Information Science and Engineering, Central South University, Changsha 410083, China; 3. Department of Physics and Electronics Information Science, Hengyang Normal College, Hengyang 421008, China. Correspondent: LONG Wen, E-mail: lw770457@163.com)

**Abstract:** A hybrid evolutionary algorithm based on gradient descends method is proposed to determine the structure of RBF neural network and optimize its parameters. A gradient descend operator is introduced into the evolutionary algorithm. Gradient descend method is carried on search by certain probability for certain elitists of every generation to strengthen the ability of local search. The structure and parameters of RBF network are trained and optimized together by using hybrid evolutionary algorithm. The experimental results show that the RBF neural network has stronger generalization ability.

**Key words:** radial basis function neural network; evolutionary algorithm; gradient descend method; nonlinear time series forecasting

### 1 引言

时间序列预测是时间序列分析中的一项主要工作, 它根据系统有限长度的历史数据建立模型, 并对系统的未来行为进行预测. 然而, 传统的时间序列预测模型一般都是线性的, 如递归最小二次方法、自回归模型、求和自回归移动平均模型等, 它们在刻画复杂的非线性系统时常常难以获得满意的预测效果<sup>[1]</sup>. 近年来, 大量的非线性模型, 如基于误差反向传播算法(BP)的人工神经网络、径向基函数(RBF)神经网络、支持向量机回归模型等, 在非线性的时间序列预测中得到了广泛的应用<sup>[2-5]</sup>. 与BP神经网络相比, RBF神经网络具有全局收敛的特点, 对于非线性系统的逼

近精度和收敛速度都较优. 然而, RBF神经网络的基函数数目、中心及宽度、连接权值的确定一直是难以解决的问题, 至今没有系统的规律可循. 因此, 一些研究者在如何确定网络结构和参数方面作了许多工作, 如Karayiannis<sup>[6]</sup>提出了基于梯度下降法训练RBF网络, 确定其网络结构和参数, 取得了较好的效果. 甘敏等人<sup>[7]</sup>采用两种混合优化算法来确定RBF网络的中心、基函数个数和权值, 在Mackey-Glass混沌时间序列预测应用中验证了优化后网络的有效性.

近年来, 由于具有较强的全局收敛能力, 以遗传算法为代表的进化算法也用来确定RBF网络的结构和优化其参数, 取得了较好的效果. 但是遗传算法收

收稿日期: 2010-12-27; 修回日期: 2011-03-07.

基金项目: 国家自然科学基金项目(60874070, 61074069); 湖南省研究生科研创新项目(CX2009B038).

作者简介: 龙文(1977-), 男, 博士生, 从事神经网络、进化计算的研究; 梁昔明(1967-), 男, 教授, 博士生导师, 从事非线性系统建模与优化、进化计算等研究.

敛速度慢,导致网络的训练时间过长<sup>[8]</sup>。为了解决这一问题,文献[9-10]将梯度下降法和遗传算法相结合,提出了两步学习方法,即先用遗传算法对RBF网络的参数进行训练得到较优解,然后再利用梯度下降法对网络参数进行进一步优化,通过将其应用到交通信息预测和UCI机器学习中表明了混合算法的有效性。

本文结合进化算法较强的全局搜索能力和梯度下降法较强的局部搜索能力,提出了一种混合进化算法。该方法采用混合编码方式对网络结构和参数同时进行编码,将训练样本误差和验证样本误差的加权作为个体的适应度函数,利用混合进化算法对RBF神经网络的结构和参数同时进行训练和优化。通过Lorenz和Mackey-Glass两个混沌非线性时间序列的预测来验证该RBF网络模型的有效性。

## 2 RBF神经网络结构

RBF神经网络的结构通常由输入层、隐含层和输出层组成,通过非线性基函数的线性组合来实现输入到输出的映射关系,其结构如图1所示。

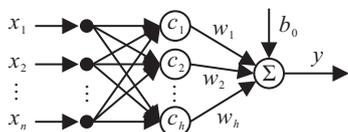


图1  $n-h-1$ 结构的RBF神经网络

图1中,RBF神经网络结构具有 $n$ 个输入, $h$ 个隐含节点和1个输出。 $X = (x_1, x_2, \dots, x_n)^T \in \mathbf{R}^n$ 为网络输入矢量, $C_i$ 为隐节点中心, $w_i (i = 1, 2, \dots, h)$ 为连接第 $i$ 个隐节点和输出节点的权值, $b_0$ 为输出单元偏移,径向基函数选取常用的高斯函数,则对于输入样本 $X$ ,第 $i$ 个隐节点的输出为

$$\varphi_i(X) = \exp\left(-\frac{\|X - C_i\|^2}{\sigma_i^2}\right), \quad (1)$$

其中 $\sigma_i$ 为第 $i$ 个基函数的宽度。RBF网络的输出为

$$Y = \sum_{i=1}^h w_i \varphi_i(X) + b_0. \quad (2)$$

从网络结构分析中可以看出,在RBF网络中需要确定的参数有两类:1)基函数宽度 $\sigma_i$ ,隐节点中心 $C_i$ 和中心数目 $h$ ;2)隐含层和输出层之间的连接权值 $w_i$ 。传统的确定RBF网络结构和参数的方法是离线两阶段学习算法。基函数的中心和宽度通常由 $K$ -均值聚类算法得到,连接权值通过梯度下降法训练得到。影响RBF网络泛化能力的因素较多,包括训练样本的数量和质量、网络结构的复杂性、网络的初始权值和训练时间等,其中网络结构的复杂性对于RBF网络的泛化能力影响极大。

## 3 基于混合进化算法的RBF神经网络

进化算法是一种通过选择、交叉和变异等机制

来实现的优化算法,具有较强的全局搜索能力,适合于RBF网络参数的优化;但参数较多时,单纯采用进化算法进行优化时则存在收敛速度慢的问题。为了提高收敛速度,本文在进化算法中嵌入梯度下降方法,结合两种方法的优点,提出一种混合进化算法对RBF网络结构和参数进行训练和优化。

### 3.1 混合编码及改进适应度

在确定隐节点中心数目 $h$ ,中心 $C_i$ 和宽度 $\sigma_i$ 后,隐层和输出层的连接权值 $w_i$ 构成一个线性方程组,采用递推最小二乘法即可求得。为了减少算法的复杂度,不是将连接权值 $w_i$ , $h$ , $C_i$ 和 $\sigma_i$ 一起混合编码,而是确定其他3个参数后再求权值。因此,确定参数 $h$ , $C_i$ 和 $\sigma_i$ 的值是建立RBF神经网络的关键。

为了实现自动选择RBF神经网络节点数目、输入变量以及同时优化其参数,本文采用混合编码方式,即对每个个体同时采用二进制和实数编码。其中:二进制部分对隐节点的编码,实数部分对网络参数中心 $C_i$ 和宽度 $\sigma_i$ 进行编码。进化算法中个体的混合编码结构如图2所示。在二进制编码部分,“1”表示选择某个节点,“0”表示不选择某个节点,则计算节点的数目就是编码相加。如“11001”表示有3个节点,实数编码部分直接用变量范围内的实数来表示。

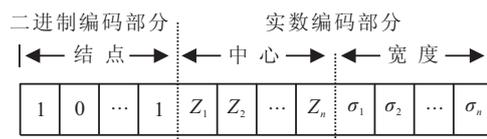


图2 个体混合编码结构

进化算法中,个体的适应度大小是衡量该个体进化到下一代种群的关键。采用进化算法优化RBF网络参数时,一般采用训练样本误差作为种群中个体的适应度函数。然而,如果过分地追求训练样本误差小,则有可能出现过度拟合现象,即在训练中实际测量值与网络输出值的误差较小;但是在测试中实际测量值与网络输出值的误差较大。受文献[11]的启发,本文对适应度函数进行改进,将训练样本误差和验证样本误差加权作为个体的适应度函数。在进行网络训练前,将样本集 $I$ 随机分为训练集 $A$ 和验证集 $B$ ,其中 $A \cap B = \emptyset$ , $A \cup B = I$ 。定义训练误差 $\text{Error}_A$ 和验证误差 $\text{Error}_B$ 分别为

$$\begin{aligned} \text{Error}_A &= \frac{1}{N_A} \sum_{j=1}^{N_A} (y_j - y_j^*)^2, \\ \text{Error}_B &= \frac{1}{N_B} \sum_{j=1}^{N_B} (y_j - y_j^*)^2. \end{aligned} \quad (3)$$

其中: $N_A$ 和 $N_B$ 分别为RBF网络训练样本集 $A$ 和验证样本集 $B$ 中的数据个数, $y_j$ 和 $y_j^*$ 分别为实际测量值和网络预测输出值。与文献[11]相似,本文RBF网

网络的优化问题是求解一组满足精度大小的参数,使

$$E = \alpha \text{Error}_A + (1 - \alpha) \text{Error}_B \quad (4)$$

最小,其中  $0 < \alpha < 1$ , 权重  $\alpha$  的大小根据实际应用中逼近精度的要求决定. 如果要求网络的验证误差相对小些, 则  $\alpha$  取较小的值, 反之, 则  $\alpha$  取较大的值. 以改进的误差函数 (4) 作为适应度函数, 能直观地反映个体性能的优劣. 式 (4) 的本质是将训练样本和测试样本训练网络得到的参数进行加权平均后可得到 RBF 网络最终的参数, 这样, 在保证训练误差达到要求的前提下, 提高了 RBF 网络反映非训练样本输入输出关系的能力, 即 RBF 网络的泛化能力.

### 3.2 梯度下降法

对于每次进化后产生的新种群, 选择其中适应度值较高的  $S$  个精英个体以概率  $p_s$  进行梯度下降迭代搜索. 梯度下降法的具体迭代过程如下:

Step 1: 给定初始点  $x^0 \in \mathbf{R}^n$ , 终止误差  $\varepsilon > 0$ , 令  $k = 0$ , 最大迭代次数为  $\max\_iter$ .

Step 2: 计算  $d^k = -\nabla f(x^k)$ , 如果  $\|d^k\| < \varepsilon$  或  $k > \max\_iter$ , 则停止,  $x^k$  为最优解.

Step 3: 采用不精确 Goldstein-Armijo 准则线性搜索得到步长因子  $\alpha^k$ .

Step 4: 令  $x^{k+1} = x^k - \alpha^k \nabla f(x^k)$ ,  $k = k + 1$ , 转至 Step 2.

选取若干个精英个体以一定概率进行梯度下降搜索的目的, 是加强算法的局部搜索能力和加快算法的收敛速度.

### 3.3 交叉和变异

交叉和变异是进化算法中的核心操作, 通过不断更新个体来增加群体的多样性. 本文对种群中不进行梯度下降搜索的个体进行单形交叉和均匀变异操作. 单形交叉算子是基于均匀分布来产生后代个体且不需要个体的适应度信息, 实现较为简单, 计算复杂度仅为  $o(N)$ , 种群规模为  $N$ . 具体操作如下: 在  $\mathbf{R}^n$  中,  $n + 1$  个独立父代个体  $x_i (i = 1, 2, \dots, n + 1)$  形成一个单形, 通过以下步骤产生后代个体:

Step 1: 按一定比率将单形向各个方向  $(x - o)$  进行扩张得到新的单形,  $o$  是  $n + 1$  个向量的中心, 即

$$o = \frac{1}{n+1} \sum_{i=1}^{n+1} x_i. \quad (5)$$

Step 2: 从新的单形中选择一个点作为后代个体, 考虑二维空间中的 3 个点  $x_1, x_2$  和  $x_3$  构成的单形, 将该单形以  $(1 + \lambda)(\lambda \geq 0)$  的比例向各方向进行扩张, 令

$$o = \frac{1}{3} \sum_{i=1}^3 x_i, y_i = (1 + \lambda)(x_i - o), \quad (6)$$

则由  $y_1, y_2$  和  $y_3$  形成了一个新的单形. 在新单形中任取一点  $z$ , 则有

$$z = \beta_1 y_1 + \beta_2 y_2 + \beta_3 y_3 + o, \quad (7)$$

其中  $\beta_1, \beta_2$  和  $\beta_3$  为  $[0, 1]$  中均匀分布的随机数且满足  $\beta_1 + \beta_2 + \beta_3 = 1$ . 单形交叉一般写为  $\text{SPX} - \mu - \rho - \lambda$ . 其中:  $\mu$  为随机选择进行交叉操作的父代个体数,  $\rho$  为产生的子代个体,  $\lambda$  为扩张比例控制参数. 单形交叉算子使算法在进化初期有较强的全局搜索能力, 在进化后期有较强的局部搜索能力.

均匀变异首先选取变异点处对应元素取值范围内的一随机数, 然后以变异概率  $p_m$  替换编码串中该变异点处的原有元素值. 具体操作过程如下:

Step 1: 依次指定个体编码串中的每个元素作为变异点.

Step 2: 对于各变异点, 以变异概率  $p_m$  从对应元素的取值范围内取一随机数替代原有元素值.

设个体  $X = x_1 x_2 \dots x_k \dots x_l$ , 若  $x_k$  为变异点, 其取值范围为  $[U_{\min}^k, U_{\max}^k]$ , 则在该点进行均匀变异操作后得到一个新的个体  $X' = x_1 x_2 \dots x'_k \dots x_l$ , 变异点的新元素值为

$$x'_k = U_{\min}^k + r(U_{\max}^k - U_{\min}^k), \quad (8)$$

其中  $r$  为  $[0, 1]$  的一个随机数. 均匀变异操作可使搜索点在整个搜索空间内自由移动, 从而增加种群的多样性, 防止算法的早熟收敛.

### 3.4 算法步骤

基于以上分析, 利用混合进化算法优化 RBF 网络结构和参数的具体步骤如下:

Step 1: 设置相关参数. 种群规模  $N$ , 精英个体  $S$ , 概率  $p_s$  和  $p_m$ .

Step 2: 个体按第 3.1 节中的编码方法进行编码, 随机初始化种群 (每个个体由多维向量组成).

Step 3: 令  $k = 1$ .

Step 4: 将每个向量映射为网络的一组参数值, 并组成网络, 输入训练样本进行训练.

Step 5: 判断是否满足结束条件, 若满足, 则输出网络最优参数值, 构建网络进行时间序列预测; 否则, 转向 Step 6.

Step 6: 由式 (4) 计算每个个体的适应度.

Step 7: 选取  $S$  个精英个体以概率  $p_s$  进行梯度下降法搜索, 种群中剩下的个体则进行单形交叉和均匀变异操作.

Step 8: 令  $k = k + 1$ , 返回 Step 4.

## 4 数值实验

为了验证本文方法的有效性, 采用 Lorenz 时间序列和 Mackey-Glass 混沌时间序列预测进行实验, 并与现存的一些其他模型或方法作比较.

### 4.1 预测 Lorenz 混沌时间序列

非线性混沌 Lorenz 系统的微分方程<sup>[12]</sup>为

$$\begin{cases} \dot{x} = -a(x - y), \\ \dot{y} = -xz + cx - y, \\ \dot{z} = xy - bz. \end{cases} \quad (9)$$

当  $c \geq 28$  时, Lorenz 系统 (9) 处于混沌状态, 此处与以前的文献保持一致, 选取  $a = 10$ ,  $b = 8/3$ ,  $c = 28$ , 初始值为  $(0.0031, 0.1928, 0.4208)$ , 采用取样时间为  $0.01 \text{ s}$  生成单变量  $x$  的数据, 如图 3 所示.

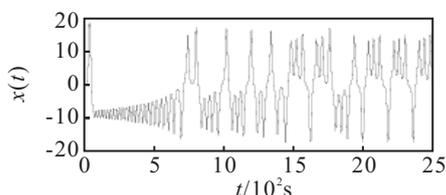


图3  $c = 28$  时的 Lorenz 混沌时间序列

用前 1500 对数据样本进行训练建立 RBF 网络模型, 后 1000 对数据样本用来测试模型的有效性. 为了方便与其他方法进行比较, 本文混合进化算法 (HEA) 的参数 (通过实验讨论得到的一组最优参数, 由于篇幅原因, 参数讨论内容在此没有详细叙述) 设置为:  $N = 30$ ,  $S = 2$ ,  $p_s = 0.5$ ,  $p_m = 0.08$ . 单形交叉中:  $\mu = 10$ ,  $\rho = 5$ ,  $\lambda = 10$ . 进化算法 (EA) 的参数设置为: 种群规模为 50, 交叉概率  $p_c = 0.7$ , 变异概率  $p_m = 0.1$ . 其他算法的参数见各自文献. 定义性能指标为归一化的均方误差 (NMSE), 有

$$\text{NMSE} = \left( \frac{\sum (x(t) - \hat{x}(t))^2}{\sum (\hat{x}(t) - \bar{x})^2} \right), \quad (10)$$

其中  $x(t)$ ,  $\hat{x}(t)$  和  $\bar{x}$  分别为测量值、预测值和平均值. 表 1 给出了本文方法 (独立运行 20 次的平均结果) 和文献中其他方法对 Lorenz 混沌时间序列的预测结果, 各种方法的预测精度以 NMSE 的形式给出. 由表 1 可知, 本文方法得到的结果要优于文献中的其他方法.

表 1 Lorenz 时间序列预测结果的比较

模型	节点数	训练样本 NMSE	测试样本 NMSE
文献 [9]	25	$6.3387 \times 10^{-9}$	$2.3511 \times 10^{-9}$
文献 [10]	49	$3.6016 \times 10^{-8}$	$5.0268 \times 10^{-8}$
本文方法	8	$1.9806 \times 10^{-11}$	$3.6249 \times 10^{-11}$

## 4.2 预测 Mackey-Glass 混沌时间序列

基于 Mackey-Glass 时滞微分方程的混沌时间序列预测, 通常被认为是检验和比较各种非线性系统模型和算法的一个标准测试问题. Mackey-Glass 时滞微分方程为

$$\frac{dx(t)}{dt} = \frac{ax(t - \tau)}{1 + x^c(t - \tau)} - bx(t), \quad (11)$$

其中  $\tau$  为时滞参数, 当  $\tau \geq 17$  时方程 (11) 呈现混沌性. 与以前文献保持一致, 选取  $a = 0.2$ ,  $b = 0.1$ ,  $c =$

$10$ ,  $\tau = 17$ , 初始值  $x(0) = 1.2$ , 采用四阶 Runge-Kutta 法生成 1000 个数据, 图 4 为  $\tau = 17$  时 Mackey-Glass 混沌时间序列.

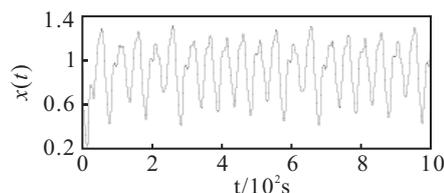


图4  $\tau = 17$  时的 Mackey-Glass 混沌时间序列

对 Mackey-Glass 混沌时间序列进行预测是指根据  $t$  时刻以前的一组数据  $[x(t), x(t - \Delta t), x(t - 2\Delta t), \dots, x(t - (l - 1)\Delta t)]$  去预测  $x(t + \Delta t)$ ,  $\Delta t$  为预测时间步幅. 选取  $l = 3$ ,  $\Delta t = 6$ , 得到 1000 组样本数据, 即  $[x(t - 18), x(t - 12), x(t - 6), x(t); x(t + 6)]$ ,  $t = 19, 20, \dots, 1018$ . 其中前 4 个数据作为输入, 最后 1 个数据作为输出. 用前 500 对数据样本进行训练建立 RBF 网络模型, 后 500 对数据样本用来测试模型的有效性.

为了表明本文算法的稳定性和普适性, 同时也方便与其他方法进行比较, 这里算法参数设置与第 4.1 节相同. 表 2 给出了本文所提出的混合进化算法与其他方法的实验结果, 包括节点数、训练样本和测试样本的均方误差 (MSE) 的平均值.

表 2 Mackey-Glass 时间序列预测结果的比较

方法	节点数	训练样本 MSE	测试样本 MSE
文献 [9]	25	$3.2027 \text{E-}05$	$2.6104 \text{E-}05$
文献 [10]	49	$2.5891 \text{E-}04$	$1.8920 \text{E-}04$
本文方法	25	$2.0538 \text{E-}06$	$1.7329 \text{E-}06$

由表 2 可知, 利用混合进化算法得到的测试样本误差比训练样本误差稍小, 这说明 HEA-RBF 网络从训练样本中获取了较好的预测能力, 其预测精度要优于文献中的其他方法.

## 5 结论

本文提出了一种基于梯度下降法的混合进化算法, 并将其用于 RBF 网络结构和参数优化. 该方法把网络节点、中心和宽度混合编码, 在进化算法中嵌入梯度下降法以增强算法的局部搜索能力和收敛速度. 对 2 个非线性混沌时间序列进行预测的结果表明, 本文方法的预测精度优于其他方法且所用节点数较少.

## 参考文献 (References)

- [1] Rojas I, Valenzuela O, Rojas F, et al. Soft-computing technique and ARMA model for time series prediction[J]. Neurocomputing, 2008, 71(4-6): 519-537.

(下转第 1272 页)