

文章编号: 1001-0920(2009)08-1172-05

## RBF 神经网络参数估计的两种混合优化算法

甘 敏<sup>1</sup>, 彭晓燕<sup>2</sup>, 彭 辉<sup>1</sup>

(1. 中南大学 信息科学与工程学院, 长沙 410083; 2. 湖南大学 机械与汽车工程学院, 长沙 410082)

**摘 要:** 基于全局搜索的进化算法和一种局部搜索算法——结构化的非线性参数优化方法(SNPOM), 提出两种混合的优化算法来估计 RBF 神经网络中的参数: 1) 初始化一定数目的种群作为 SNPOM 的初始值得到其适应值, 通过选择、交叉和替换策略来更新种群; 2) 采用进化算法运行一定的代数, 从最终群体中选取一些个体进一步用 SNPOM 来优化. 这两种混合优化算法的本质是用进化算法为 SNPOM 搜寻最优初始值, 以得到全局最优解. 仿真实验结果表明, 该混合算法比单独使用进化算法或 SNPOM 更优, 且优于其他一些算法.

**关键词:** RBF 神经网络; 参数估计; 混合优化方法

**中图分类号:** TP18

**文献标识码:** A

## Two hybrid parameter optimization algorithms for RBF neural networks

GAN Min<sup>1</sup>, PENG Xiao-yan<sup>2</sup>, PENG Hui<sup>1</sup>

(1. School of Information Science and Engineering, Central South University, Changsha 410083, China; 2. College of Mechanical and Automobile Engineering, Hu'nan University, Changsha 410082, China. Correspondent: GAN Min, E-mail: ganmin@mail.csu.edu.cn)

**Abstract:** Based on an evolutionary algorithm (EA) and a local search strategy — the structured nonlinear parameter optimization method (SNPOM), two hybrid parameter optimization algorithms for RBF neural networks are proposed. The first approach starts with a population of some random initial parameter values, and updates the population by selection, crossover and replacement according to the fitness values obtained by using SNPOM. The second method runs the EA for a reasonable amount of generations, after which the SNPOM is used to locate the refined local optimum. The basic idea of the two hybrid algorithms is to find the optimal initial values for SNPOM using EA. The simulation tests show that the combination method provides better results than either the single method (EA and SNPOM) or some other existing algorithms.

**Key words:** RBF neural network; Parameter estimation; Hybrid optimization approach

### 1 引 言

RBF 神经网络以其简单的结构、较高的非线性逼近精度和快速学习的能力, 广泛地应用于模式识别、函数逼近、信号处理、时间序列预测等领域<sup>[1-7]</sup>, 已成为最流行的前馈神经网络之一. 然而, 由于目标函数包含大量的局部最优点, 估计神经网络模型中的参数(或训练神经网络)是非常困难的优化问题. 一般而言, RBF 神经网络包含 3 类参数: RBF 网络中心、宽度和线性权重. 典型的参数优化(或学习)过程分为两个步骤: 1) 非线性优化用来选择中心和宽度; 2) 线性优化用来确定线性权重. 研究表明, RBF

网络中心的选择对模型的性能具有重要的影响, 这也是优化的主要任务之一<sup>[4]</sup>.

最常用的方法是基于进化算法或基于梯度的方法. Shi 等<sup>[2]</sup>提出用进化规划算法来优化 RBF 网络中心. Du 等<sup>[7]</sup>提出一种新的遗传编码策略, 用来优化 RBF 网络的节点数、输入变量以及中心和宽度. Panchapakesan 等<sup>[4]</sup>用基于梯度的搜索方法来优化 RBF 网络中心. 彭辉等<sup>[5]</sup>提出一种结构化的非线性参数优化方法(SNPOM), 该方法将被搜索参数空间分为非线性参数空间和线性参数空间, 用类似于 Levenberg-Marquardt 方法(LMM)来优化中心, 用

收稿日期: 2008-10-22; 修回日期: 2009-01-13.

基金项目: 国家自然科学基金项目(60443008); 湖南省自然科学基金项目(07JJ3126).

作者简介: 甘敏(1982—), 男, 武汉人, 博士生, 从事神经网络、进化计算的研究; 彭辉(1963—), 男, 长沙人, 教授, 博士生导师, 从事建模与辨识、参数优化等研究.

最小二乘法 (LSM) 来优化线性权重, 使收敛性得到极大的提高. 然而, 这些方法都是单独使用随机搜索或基于梯度搜索的方法, 得到的解往往是局部最优解.

对于某些关键问题, 如能搜索到或接近于全局最优解, 将有重要的意义. Quagliarella 等<sup>[8]</sup>指出, 结合基于梯度的方法与进化算法的混合方法, 可大大增强得到全局最优解的可能性, 取得比单独使用二者更好的结果. 为此, 本文基于进化算法和 SNPOM, 提出两种混合的优化算法, 用于估计 RBF 神经网络中的参数. 其本质是用进化算法来搜索 SNPOM 的最优初始值, 以寻求全局最优解. 将这两种混合算法的性能与其他算法的性能相比较, 辨识出的 RBF 神经网络用来预测一种标准的时间序列. 实验结果表明, 混合算法具有较高的性能, 且优于现有的其他算法.

## 2 RBF 神经网络结构

RBF 神经网络是三层的前馈网络, 包括输入层、隐含层和输出层. 图 1 给出了只含一个输出结点的 RBF 网络框架. 其输入输出关系可表示为

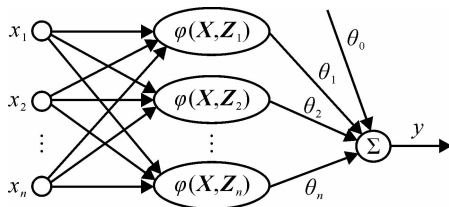


图 1 RBF 神经网络结构

$$y = \theta_0 + \sum_{i=1}^n \theta_i \varphi(\mathbf{X}, \mathbf{Z}_i). \quad (1)$$

其中:  $\theta_0$  称为偏差,  $\theta_i (i = 1, 2, \dots, n)$  为连接隐含结点和输出结点的常量权重,  $\varphi(\mathbf{X}, \mathbf{Z}_i)$  为第  $i$  个结点中输入向量  $\mathbf{X}$  与其中心  $\mathbf{Z}_i$  的距离,  $n$  为隐含层结点数.

RBF 函数的类型有多种, 本文采用最常用的高斯径向基函数, 即

$$\varphi(\mathbf{X}, \mathbf{Z}) = \exp\left(-\frac{1}{\sigma^2} \|\mathbf{X} - \mathbf{Z}\|^2\right). \quad (2)$$

其中:  $\sigma$  称为宽度,  $\|\cdot\|$  表示向量的欧式距离. 结合方程(1)和(2), 高斯 RBF 神经网络可表示为

$$y = \theta_0 + \sum_{i=1}^n \theta_i \exp\left(-\frac{1}{\sigma_i^2} \|\mathbf{X} - \mathbf{Z}_i\|^2\right). \quad (3)$$

## 3 RBF 神经网络的两种混合优化算法

估计 RBF 神经网络中的参数是很困难的. 由彭辉等<sup>[5]</sup>提出的 SNPOM 虽有收敛快、精度高的优良性能, 但基于梯度的优化方法强烈地依赖于初始值的选取. 对于神经网络这种存在大量局部最优解的

优化问题, SNPOM 极有可能落入局部最优解. 更好的优化途径之一可能是全局搜索与局部搜索相结合的方法<sup>[8,9]</sup>.

作为一种全局搜索的方法, 进化算法具有目标函数不需可微或连续、易执行、稳健等优点, 常被用来求解含有大量局部最优点的优化问题. 最近, Deb<sup>[10]</sup>提出一种实数编码的基于群的算法发生器. 该方法把搜索分为 4 个独立的策略: 1) 选择策略; 2) 生成后代策略; 3) 替换策略; 4) 更新策略. 受文献<sup>[10]</sup>的启发, Cai 等<sup>[11]</sup>设计出一种高效的基于群的算法发生器模型. 本文基于类似于文献<sup>[11]</sup>的进化模型和 SNPOM, 提出两种混合的优化算法, 用于估计 RBF 网络中的参数.

### 3.1 第 1 种混合优化算法

局部搜索算法与进化算法相结合的方法有多种, 本文所用的第 1 种方法是 Lamarckian 方法<sup>[9]</sup>: 一些新产生的个体用于局部搜索算法, 从中选出最好的局部最优解来替换当前群体中的个体. 该算法的框图如图 2 所示.

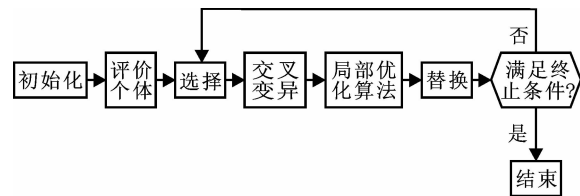


图 2 第 1 种混合优化方法框图

采用如上方法对 RBF 神经网络参数进行优化, 算法的具体执行步骤如下:

第 1 步: 参数分类. RBF 神经网络(3)的参数可分为线性参数

$$\boldsymbol{\theta}_L \triangleq (\theta_0, \theta_1, \dots, \theta_n)^T, \quad (4)$$

以及非线性参数

$$\boldsymbol{\theta}_N \triangleq (\sigma_1, \sigma_2, \dots, \sigma_n, \mathbf{Z}_1^T, \mathbf{Z}_2^T, \dots, \mathbf{Z}_n^T)^T. \quad (5)$$

为了表述方便, 把模型(3)改写成如下两种形式:

$$y(t) = f(\boldsymbol{\theta}_L, \boldsymbol{\theta}_N, \mathbf{X}(t-1)) + e(t), \quad (6)$$

$$y(t) = \boldsymbol{\Psi}(\boldsymbol{\theta}_N, \mathbf{X}(t-1))^T \boldsymbol{\theta}_L + e(t). \quad (7)$$

显然, 式(7)是式(6)线性参数的回归形式.

第 2 步: 初始化. 初始化规模为  $N$  的群体(集合  $P$ ), 表示可能的 RBF 网络中心向量, 记为  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ .

第 3 步: 用 SNPOM 计算个体的目标函数值.

1) 把个体  $\mathbf{x}_i$  改写成  $\mathbf{Z}_k^0 (k = 1, 2, \dots, n)$  的形式, 作为 RBF 网络的初始中心, 并用下式来计算宽度的初始值:

$$\sigma_k^0 = \sqrt{\max_{t-1} \{ \|\mathbf{X}(t-1) - \mathbf{Z}_k\|_2^2 \} / (-\log \epsilon_k)}, \quad \epsilon_k \in [0.1 \sim 0.0001]. \quad (8)$$

这样作是为保证:当  $\mathbf{X}(t-1)$  离中心  $\mathbf{Z}_k$  很远时,  $\exp\{-\lambda_k \|\mathbf{X}(t-1) - \mathbf{Z}_k\|_2^2\}$  项趋近于 0,使线性权重稳定有界<sup>[2,5]</sup>. 选择了非线性参数后,用 LSM 来计算线性参数

$$\begin{cases} \boldsymbol{\theta}_L^0 = \left[ \sum_{t=\tau}^M \boldsymbol{\Psi}_{t,0} \boldsymbol{\Psi}_{t,0}^T \right]^{-1} \sum_{t=\tau}^M \boldsymbol{\Psi}_{t,0} \bar{y}(t), \\ \boldsymbol{\Psi}_{t,0} = \boldsymbol{\Psi}(\boldsymbol{\theta}_N^0, \bar{\mathbf{X}}(t-1)), i = \tau + 1, \dots, M. \end{cases} \quad (9)$$

其中:  $\bar{y}(i)$  和  $\bar{\mathbf{X}}(t-1)$  是观测数据集,  $\tau$  是模型的最大时间延迟,  $M$  是观测数据的个数.

2) 定义优化目标函数为

$$\begin{aligned} V(\boldsymbol{\theta}_N, \boldsymbol{\theta}_L) &= \frac{1}{2} \|\mathbf{F}(\boldsymbol{\theta}_N, \boldsymbol{\theta}_L)\|_2^2, \quad (10) \\ \mathbf{F}(\boldsymbol{\theta}_N, \boldsymbol{\theta}_L) &= \begin{bmatrix} f(\boldsymbol{\theta}_N, \boldsymbol{\theta}_L, \bar{\mathbf{X}}(\tau)) - \bar{y}(\tau+1) \\ f(\boldsymbol{\theta}_N, \boldsymbol{\theta}_L, \mathbf{X}(\tau+1)) - \bar{y}(\tau+2) \\ \vdots \\ f(\boldsymbol{\theta}_N, \boldsymbol{\theta}_L, \bar{\mathbf{X}}(M-1)) - \bar{y}(M) \end{bmatrix}. \end{aligned} \quad (11)$$

因此,参数优化问题为

$$(\hat{\boldsymbol{\theta}}_n, \hat{\boldsymbol{\theta}}_L) = \arg \min_{\boldsymbol{\theta}_N, \boldsymbol{\theta}_L} V(\boldsymbol{\theta}_N, \boldsymbol{\theta}_L). \quad (12)$$

3) 两个迭代过程用来优化所要估计的参数. 对于非线性参数向量  $\boldsymbol{\theta}_N$ , 更新公式为

$$\boldsymbol{\theta}_N^{k+1} = \boldsymbol{\theta}_N^k + \beta_k \mathbf{d}_k. \quad (13)$$

其中:  $k$  表示迭代步数,  $\mathbf{d}_k$  为搜索方向,  $\beta_k$  为步长系数. 对于线性参数向量  $\boldsymbol{\theta}_L$ , 与式(9)相似, 有

$$\begin{cases} \boldsymbol{\theta}_L^{k+1} = \left[ \sum_{t=\tau}^M \boldsymbol{\Psi}_{t,k+1} \boldsymbol{\Psi}_{t,k+1}^T \right]^{-1} \sum_{t=\tau}^M \boldsymbol{\Psi}_{t,k+1} \bar{y}(t), \\ \boldsymbol{\Psi}_{t,k+1} = \boldsymbol{\Psi}(\boldsymbol{\theta}_N^{k+1}, \bar{\mathbf{X}}(t-1)). \end{cases} \quad (14)$$

在式(13)中, 每次迭代  $\mathbf{d}_k$  由如下方程决定:

$$\begin{cases} [\mathbf{J}(\boldsymbol{\theta}_N^k)^T \mathbf{J}(\boldsymbol{\theta}_N^k) + \gamma_k \mathbf{I}] \mathbf{d}_k = -\mathbf{J}(\boldsymbol{\theta}_N^k)^T \mathbf{F}(\boldsymbol{\theta}_N^k, \boldsymbol{\theta}_L^k), \\ \mathbf{J}(\boldsymbol{\theta}_N^k) = (\partial \mathbf{F}(\boldsymbol{\theta}_N^k, \boldsymbol{\theta}_L^k) / \partial \boldsymbol{\theta}_N^k)^T. \end{cases} \quad (15)$$

其中  $\gamma_k$  控制  $\mathbf{d}_k$  的幅度和方向. 当  $\gamma_k \rightarrow 0$  时,  $\mathbf{d}_k$  将沿高斯-牛顿方向; 当  $\gamma_k \rightarrow \infty$  时,  $\mathbf{d}_k$  将沿最速下降方向. 当  $\mathbf{d}_k$  求得后, 式(13)中的  $\beta_k$  由类似于混合二次和三次多项式内插和外插法的线搜索过程得到. 在优化过程中, 每当 RBF 中心  $\boldsymbol{\theta}_N^k$  更新后,  $\sigma_k$  都要用式(8)进行更新.

第 4 步: 选择. 从群体  $P$  中随机选取  $\mu$  个个体(集合  $Q$ ),  $P = P - Q$ .

第 5 步: 生成后代. 用单纯形杂交算子把集合  $Q$  中的个体作为父体, 产生  $\lambda$  个后代(集合  $R$ ).

第 6 步: 替换. 用第 3 步中的方法计算集合  $R$  中个体的目标函数值, 并选出目标函数值最小的个体, 记为  $s_1$ ; 从集合  $Q$  中选出目标函数值最大的个体, 记

为  $s_2$ . 如果  $V(s_1) < V(s_2)$ , 则用  $s_1$  替换  $s_2$ ; 否则, 没有替换.

第 7 步: 更新.  $P = P \cup Q$ .

第 8 步: 重复第 4 步 ~ 第 7 步, 直到满足终止条件.

### 3.2 第 2 种混合优化算法

第 2 种优化方法采用文献[8]建议的混合方式: 先用进化算法运行一定数目的代数, 再从群体中选择一些个体作为局部搜索算法的初始值进行优化. 其算法框图如图 3 所示.

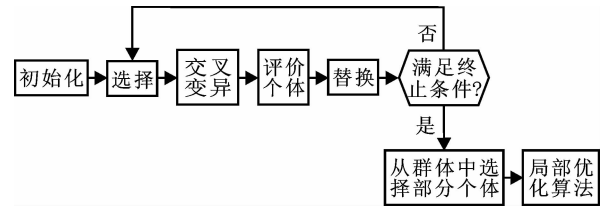


图 3 第 2 种混合优化方法框图

这种 RBF 神经网络优化算法的具体执行步骤如下:

第 1 步: 初始化. 初始化规模为  $N$  的群体(集合  $P$ ), 表示可能的 RBF 网络中心向量, 记为  $x_1, x_2, \dots, x_N$ . 用式(8)和(9)计算宽度和线性权重, 并把模型的均方误差作为各个体的目标函数值  $V(x_i), i = 1, 2, \dots, N$ .

第 2 步: 选择. 从群体  $P$  中随机选取  $\mu$  个个体(集合  $Q$ ),  $P = P - Q$ .

第 3 步: 生成后代. 用单纯形杂交算子把集合  $Q$  中的个体作为父体, 产生  $\lambda$  个后代(集合  $R$ ).

第 4 步: 替换. 用式(8)和(9)计算集合  $R$  中个体的宽度和线性权重, 并把模型的均方误差作为各个体的目标函数值, 选出目标函数值最小的个体, 记为  $s_1$ ; 从集合  $Q$  中选出目标函数值最大的个体, 记为  $s_2$ . 如果  $V(s_1) < V(s_2)$ , 则用  $s_1$  替换  $s_2$ ; 否则, 没有替换.

第 5 步: 更新.  $P = P \cup Q$ .

第 6 步: 重复第 2 步 ~ 第 5 步, 直到满足终止条件.

第 7 步: 从最终群体中选出  $m$  个个体, 作为 SNPOM 的初始值来优化参数(参见第 1 种优化算法第 3 步).

### 3.3 中心数目的确定

以上两种混合优化步骤是在给定 RBF 中心数情况下的估计步骤. 对于中心个数  $n$ , 可用 Akaike 信息准则(AIC)来确定<sup>[2,5,12]</sup>. 具体方法是选择不同的  $n$  值, 用 SNPOM 优化过程得到其 AIC 值, 并从中选出具有较小 AIC 值和较好动态性能的模式. 对于模

型(3), AIC 可计算如下:

$$AIC = M \log \hat{\sigma}_e^2 + 2(d + 1). \quad (16)$$

其中:  $\hat{\sigma}_e^2$  是模型的残差方差,  $d$  是所要估计的参数总数.

### 4 数值实验及分析

为检验本文所提出算法的性能, 辨识出的 RBF 网络用来预测 Mackey-Glass 时间序列. 基于 Mackey-Glass 微分方程的时间序列预测, 被认为是检验和比较各种算法性能的标准测试问题. Mackey-Glass 微分方程为

$$\frac{dy}{dt} = \frac{ay(t-\tau)}{1+y^c(t-\tau)} - by(t). \quad (17)$$

为与大多数文献保持一致, 选取  $a = 0.2, b = 0.1, c = 10, \tau = 17$ . 用 4 阶 Runge-Kutta 方法来生成数据, 初始值为  $y(0) = 1.2$ . 与文献[1, 6, 7] 相同, 选取  $t = 124 \sim 1123$  的 1000 个数据, 前 500 个数据用来训练 RBF 网络, 后 500 个数据用来检验辨识出的 RBF 网络. 为与以前的算法作公平比较, 使用固定的输入变量向量

$$[y(t-18), y(t-12), y(t-6), y(t)]$$

来预测  $y(t+6)$ . Mackey-Glass 时间序列如图 4 所示.

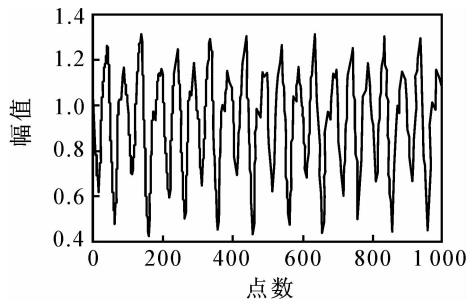


图 4 Mackey-Glass 时间序列

表 1 给出了运行两种混合优化算法和单独使用本文进化模型的参数值. 其中第 2 种混合优化算法从最终群体中选取的个体数目  $m = 40$ .

表 1 运行各方法的参数

方 法	群体规模 $N$	参数 $\mu$	参数 $\lambda$	运行代数
第 1 种混合算法	30	5	5	30
第 2 种混合算法	150	20	10	750
进化模型	150	20	10	6000

表 2 给出了每种算法运行 10 次训练数据和检验数据预测均方误差(MSE)的平均值和标准差(括号内数据), 以及它们的平均运行时间. 各算法在检验数据上得到的预测误差要比训练数据稍小, 这说明 RBF 网络从训练数据中获取了良好的预测能力. 两种混合优化算法都取得了高精度的预测效果, 比单独使用进化算法要好一个数量级, 小的标准方差

表 2 各方法的运行结果

方 法	结点数	训练数据 MSE	测试数据 MSE	运行时间 /min
第 1 种混合算法	25	2.0623E-6 (9.2138E-8)	1.7555E-6 (1.1023E-7)	89
第 2 种混合算法	25	2.1342E-6 (2.3362E-7)	1.7975E-6 (2.6173E-7)	29
进化模型	25	2.1050E-5 (1.7727E-6)	2.0045E-5 (1.9547E-6)	12

反映了算法的稳健性. 第 1 种混合算法的标准差最小, 说明其稳健性最好, 但其运行时间最长.

图 5 和图 6 给出了所提出的两种混合优化算法与 SNPOM 和进化算法的收敛性比较. 其中进化算法每隔 15 代记录一次群体平均值. 从图中可以看出, 混合方法优于单独使用 SNPOM 或进化算法.

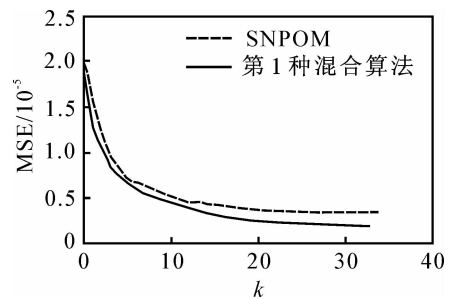


图 5 第 1 种混合优化算法与 SNPOM 收敛性比较

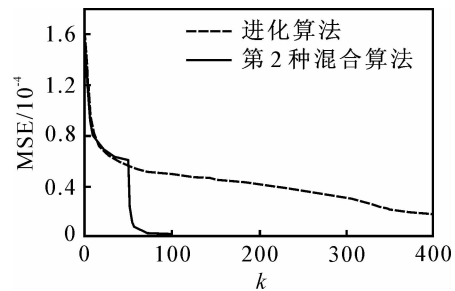


图 6 第 2 种混合优化算法与进化算法收敛性比较

表 3 给出了现有一些文献对 Mackey-Glass 时间序列的预测结果. 预测精度以均方根误差(RSME)的形式给出. 对比表 3 中各结果, 本文提出的两种混

表 3 Mackey-Glass 时间序列预测各种方法的比较

方 法	结点数	训练数据 RSME	测试数据 RSME
文献[1]	23	0.0096	0.0114
文献[3]	12	0.0029	—
文献[13]	10	0.0033	0.0036
文献[6]	116	—	0.0015
文献[7]	98	0.0014	0.0015
第 1 种混合算法	25	0.0014	0.0013
第 2 种混合算法	25	0.0015	0.0013

合优化算法得到的预测精度最高. 文献[6,7]虽然得到了相似的结果,但所用的结点数为116和98,而本文的两种混合优化算法只用25个结点便得到相似或更好的结果.因此本文所提出的混合算法明显优于其他算法.

为验证噪声对所提出算法的影响,在以上Mackey-Glass时间序列中引入标准差为0.01和0.05的高斯白噪声.表4给出了两种算法运行10次的估计结果(括号内为标准差).可以看出,两种方法得到了非常相似的结果,且都有很强的稳健性.

表4 Mackey-Glass时间序列加入噪声后的预测结果

方 法	结点数	噪声标准差	训练数据 MSE	测试数据 MSE
第1种混合算法	25	0.01	1.7742E-4 (9.5568E-7)	3.0286E-4 (5.4247E-6)
第2种混合算法	25	0.01	1.7942E-4 (2.4458E-6)	2.9939E-4 (4.4645E-6)
第1种混合算法	25	0.05	4.4020E-3 (6.0628E-5)	5.7178E-3 (2.7660E-4)
第2种混合算法	25	0.05	4.4441E-3 (4.0258E-5)	5.3965E-3 (1.5839E-4)

## 5 结 论

本文基于进化模型和SNPOM方法,提出两种混合的优化算法来估计RBF网络中的参数.从对标标准时间序列的预测结果看:1)两种混合方法都得到了较高的预测精度,且优于单独使用进化算法或SNPOM,其中第2种混合优化算法所用时间更少;2)在达到相似精度的情况下,本文提出的混合算法所需结点数大大少于其他一些算法.

## 参考文献(References)

[1] Cho K B, Wang B H. Radial basis function based adaptive fuzzy systems and their applications to system identification and prediction [J]. *Fuzzy Sets and Systems*, 1996, 83(3): 325-339.  
 [2] Shi Z, Tamura Y, Ozaki T. Nonlinear time series modeling with the radial basis function-based state-dependent autoregressive model[J]. *Int J of System*

*Science*, 1999, 30(7): 717-727.  
 [3] Rojas I, Pomares H, Bernier J L, et al. Time series analysis using normalized PG-RBF network with regression weights[J]. *Neurocomputing*, 2002, 42(1-4): 267-285.  
 [4] Panchapakesan C, Palaniswami M, Ralph D, et al. Effects of moving the centers in an RBF network[J]. *IEEE Trans on Neural Networks*, 2002, 13(6): 1299-1307.  
 [5] Peng H, Ozaki T, Haggan O V, et al. A parameter optimization method for the radial basis function type models[J]. *IEEE Trans on Neural Networks*, 2003, 14(2): 432-438.  
 [6] Harpham C, Dawson C W. The effect of different basis functions on a radial basis function network for time series prediction: A comparative study [J]. *Neurocomputing*, 2006, 69(16): 2161-2170.  
 [7] Du H, Zhang N. Time series prediction using evolving radial basis function networks with new encoding scheme [J]. *Neurocomputing*, 2008, 71(7-9): 1388-1400.  
 [8] Quagliarella D, Vicini A. Coupling genetic algorithms and gradient based optimization techniques[M]. West Sussex: John Wiley and Sons, 1998.  
 [9] Xu Y, Li G, Wu Z. A novel hybrid genetic algorithm using local optimizer based on heuristic pattern move [J]. *Applied Artificial Intelligence*, 2001, 15(7): 601-631.  
 [10] Deb K. A population-based algorithm-generator for real-parameter optimization [J]. *Soft Computation*, 2005, 9(4): 236-253.  
 [11] Cai Z, Wang Y. A multiobjective optimization-based evolutionary algorithm for constrained optimization [J]. *IEEE Trans on Evolutionary Computation*, 2006, 10(6): 658-675.  
 [12] Akaike H. A new look at the statistical model identification[J]. *IEEE Trans on Automatic Control*, 1974, 19(6): 716-723.  
 [13] Chen Y, Yang B, Dong J. Time series prediction using a local linear wavelet neural network [J]. *Neurocomputing*, 2006, 69(4-6): 449-465.