

文章编号:1001-9081(2009)09-2454-05

自适应变系数粒子群—径向基神经网络模型在负荷预测中的应用

师彪¹,李郁侠¹,于新花²,李娜¹,闫旺¹,孟欣¹

(1. 西安理工大学 水利水电学院,西安 710048; 2. 青岛科技大学 高职职业技术学院,山东 青岛 261000)

(biaosh2008@163.com)

摘要:为了提高短期电力负荷预测精度,提出了一种自适应变系数粒子群—径向基函数神经网络混合优化算法(AVCPSO-RBF)。实现了径向基神经网络参数优化。建立了基于该优化算法的短期负荷预测模型,利用贵州电网历史数据进行短期负荷预测。仿真表明,该方法的收敛速度和预测精度优于传统径向基神经网络方法和粒子群—RBF神经网络方法及基于混沌理论的神经网络模型,该优化算法克服了径向基神经网络和传统的粒子群优化方法的缺点,改善了径向基神经网络的泛化能力,提高了贵州电网短期负荷预测的精度,各日预测负荷的平均百分比误差可控制在1.7%以内。该算法可有效用于电力系统的短期负荷预测。

关键词:短期负荷预测;自适应变系数粒子群;泛化能力;径向基神经网络

中图分类号: TM614; TP18 **文献标志码:** A

Short-term load forecasting method based on neural network hybrid algorithm of adaptive variable coefficients particle swarm optimization and radial basis function

SHI Biao¹, LI Yu-xia¹, YU Xin-hua², LI Na¹, YAN Wang¹, MENG Xin¹

(1. Institute of Water Resources and Hydro-electric Engineering, Xi'an University of Technology, Xi'an Shaanxi 710048, China;

2. Technical Institute of High Vocation, Qingdao University of Science and Technology, Qingdao Shandong 261000, China)

Abstract: To improve short-term load forecasting accuracy, a neural networks hybrid optimization algorithm of Adaptive Variable Coefficients Particle Swarm Optimization and Radial Basis Function (AVCPSO-RBF) was proposed. The RBF neural network parameters could be optimized. The short-term load forecast model was established based on the AVCPSO-RBF algorithm. Using the method and history load data of Guizhou power system, the short-term load forecasting was carried out. The experimental results show that convergence of the method is faster and forecast accuracy is more accurate than that of the traditional RBF neural network algorithm, the PSO and RBF neural networks algorithm and the neural networks model based on chaos theory. The hybrid algorithm improves the RBF neural network generalization capacity, and overcomes the shortcomings of the traditional PSO algorithm and the RBF neural networks. The short-term load-forecasting accuracy is improved in Guizhou power system, of which the average percentage error is no more than 1.7%. The hybrid algorithm can be effectively used in short time load forecasting of the power system.

Key words: short-term load forecasting; Adaptive Variable Coefficients Particle Swarm Optimization (AVCPSO); generalization capacity; Radial Basis Function (RBF) neural network

0 引言

电力负荷的准确预测对于电力生产和电网安全运行以及国民经济都有着重要意义^[1]。多年来负荷预测一直是国内外研究的热点,前人在此领域已取得一定成果^[2]。传统的预测方法如时间序列法、回归分析法和模式识别法等都存在各自的缺陷。

径向基(Radial Basis Function, RBF)神经网络具有较好的非线性和自学习能力,简单易行,收敛速度快,但也存在缺点:获取数据成本很高,数据中往往包含噪声,不是很可靠^[3];隐含层神经元数常常凭经验确定,无理论依据;k均值聚类算法的结果对初始条件很敏感,常造成一部分节点学习不充分。

粒子群优化算法(Particle Swarm Optimization, PSO)具有依赖的经验参数较少、收敛速度快等优点,但也存在易陷于局部极值、收敛精度不高的缺陷。本文提出一种自适应变系数

粒子群优化算法(Adaptive Variable Coefficients Particle Swarm Optimization, AVCPSO),并用此算法训练径向基函数神经网络,实现RBF神经网络参数自动优化,建立了基于AVCPSO-RBF算法的短期电力负荷预测模型。应用该模型对贵州电网负荷进行了仿真计算,并与基于混沌理论的神经网络模型的负荷预测结果比较,证明了基于AVCPSO-RBF算法的短期电力负荷预测模型的稳定性优于传统的PSO-RBF模型、RBF网络模型和基于混沌理论的神经网络模型,而且预测精度高,各日预测电力负荷的平均百分比误差可控制在1.7%以内。网络收敛更快,并能加强RBF网络的泛化能力;为短期电力负荷预测提供了一种科学、有效的方法。

1 PSO算法及其改进

1.1 基本粒子群优化算法

粒子群优化算法是Eberhart R. C.和Kennedy J. (1995)提出的基于群智能的全局优化技术^[4],它通过粒子间的相互作

收稿日期:2009-03-05;修回日期:2009-05-25。 基金项目:国家火炬计划创新基金资助项目(07C26213711606);陕西省自然科学基金基础研究计划项目(SJ08E220);山东省软科学基金资助项目(2007RKB188)。

作者简介:师彪(1975-),男,甘肃定西人,高级工程师,博士,主要研究方向:电网规划及电力市场;李郁侠(1952-),男,陕西杨凌人,教授,博士生导师,博士,主要研究方向:电网规划及电力市场;于新花(1976-),女,山东青岛人,副教授,硕士,主要研究方向:电力自动化。

用,对解空间进行智能搜索,从而发现最优解。

设在一个 D 维的目标搜索空间中,有 N 个粒子组成一个群落,其中第 i 个粒子代表一个 D 维向量 $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$,即第 i 个粒子在 D 维搜索空间中的位置。而每个粒子的位置 x ,为一个潜在的解。将 x 代入一个目标函数即可算出其适应值,根据适应值的大小衡量 x 是否为所要求解的最优解。第 i 个粒子的“飞翔”速度也是一个 D 维向量,记为 $v_i = (v_{i1}, v_{i2}, \dots, v_{id})$ 。记第 i 个粒子到第 h 次迭代为止搜索到的最优位置为 $P_i = (P_{i1}, P_{i2}, \dots, P_{id})$,整个粒子群到第 h 次迭代为止搜索到的最优位置为 $P_{gd} = (P_{g1}, P_{g2}, \dots, P_{gd})$ 。算法的基本式如下:

$$v_{id}^{(k+1)} = \omega v_{id}^{(k)} + c_1 r_1^{(k)} (P_{id}^{(k)} - x_{id}^{(k)}) + c_2 r_2^{(k)} (P_{gd}^{(k)} - x_{id}^{(k)}) \quad (1)$$

$$x_{id}^{(k+1)} = x_{id}^{(k)} + v_{id}^{(k+1)} \quad (2)$$

其中: c_1, c_2 是加速系数,分别调节向全局最好粒子和个体最好粒子方向飞行的最大步长; r_1, r_2 为 $[0, 1]$ 的随机数; ω 为惯性因子,是控制速度的权重,一般^[5]将 ω 初始取 0.9 并使其随迭代次数的增加递减至 0.1,可以侧重于全局搜索,使搜索空间快速收敛于某区域,然后采用局部精细搜索以获得高精度的解。

式(1)等号右方第 1 部分可以理解为粒子先前的速度或惯性;第 2 部分可以理解为粒子的“认知”行为,表示粒子本身的思考能力;第 3 部分理解为粒子的“社会”行为,表示粒子之间的信息共享与相互合作。粒子在解空间内不断跟踪个体极值与全局极值进行搜索,直到达到规定的最大迭代次数或小于规定的误差标准为止。

PSO 算法操作简单,使用方便,收敛速度快。但对式(1)第 2 部分和第 3 部分分析,在算法后期,粒子群中所有粒子都聚集到一个极值点附近。此时, P_{id}, P_{gd} 与 X_{id} 相差很小,算法也存在以下缺点。

1) 粒子都是根据全体粒子和自身的搜索经验向最优解的方向“飞行”,在较大的惯性因子作用下,粒子有可能会缺乏对最优解的精细搜索而导致搜索精度不高;

2) 所有粒子都向最优解的方向“飞行”,越接近最优粒子,其速度越小,所以粒子群趋向同一,失去粒子间解的多样性,因而易于收敛到局部最优。

1.2 改进的粒子群算法

1.2.1 改进方法

AVCP SO 算法的思想为:为改善 PSO 算法的全局搜索能力,必须避免 PSO 算法后期粒子的飞行速度过小。本文在分析粒子群基本优化法的基础上,提出的动态自适应变系数粒子群优化算法。其基本原理是:粒子飞行的方向与原算法一样,都是指向两个“极值”,但新一代粒子飞行速度的大小与基本粒子群优化算法不一样,它不是由该粒子与最优粒子之间距离的大小决定,而是采用自适应递减因子或自适应递增因子修正位置变化量。即自适应递减因子或自适应递增因子修正式(1)的第(3)、(4)项位置变化量,进而自适应调整下代粒子飞行速度。其方法是:若 i 粒子前后两次飞行方向一致,需加大 i 粒子下代的速度,以加快算法的收敛速度,即自适应递增因子乘以 i 粒子上代位置变化量;若 i 粒子前后两次飞行方向不一致,需减小 i 粒子下代的速度,以提高搜索精度,即自适应递减因子乘以 i 粒子上代位置变化量。从而有效地避免在算法后期粒子群中所有粒子聚集到一个极值点附近,陷

入局部最优和早熟现象;同时,利用动态自适应惯性权值,将惯性权值表示为反映粒子群多样性的平均适应度方差的函数。避免粒子群寻优过程中表现出强烈的“趋同性”,调整式(1)的第(1)项,提高算法的收敛精度。计算式如下:

$$\begin{cases} v_{id}^{(k+1)} = \omega v_{id}^{(k)} + c_1 \gamma_1 \Delta_1 P_{id}^{(k)} + c_2 \gamma_2 \Delta_2 P_{gd}^{(k)} \\ x_{id}^{(k+1)} = x_{id}^{(k)} + v_{id}^{(k+1)} \\ |v_{id}^{(k)}| \leq V_{\max} \end{cases} \quad (3)$$

其中:

$$\begin{cases} \Delta_1 P_{id}^{(k)} = \Delta_{1id}^{(k)} \times \text{sign}(P_{id}^{(k)} - x_{id}^{(k)}) \\ \Delta_2 P_{gd}^{(k)} = \Delta_{2id}^{(k)} \times \text{sign}(P_{gd}^{(k)} - x_{id}^{(k)}) \end{cases} \quad (4)$$

其中: ω 为惯量因子; P_{id} 为粒子 i 以不同速度所经历的第 d 维最好位置分量; P_{gd} 为目前粒子群以不同速度在解空间中所经历的最好位置; $P_{id}^{(k)}$ 为第 k 代粒子 i 以不同速度所经历的第 d 维最好位置分量; $P_{gd}^{(k)}$ 为目前粒子群以不同速度在解空间中所经历的最好位置, V_{\max} 为粒子飞行的最大速度。

$\text{sign}(\ast)$ 为符号函数。其余符号同前。如果 $(P_{id}^{(k)} - x_{id}^{(k)}) \times (P_{id}^{(k-1)} - x_{id}^{(k-1)}) > 0$ 或 $(P_{gd}^{(k)} - x_{id}^{(k)}) \times (P_{gd}^{(k-1)} - x_{id}^{(k-1)}) > 0$ 说明 i 粒子前后两次飞行方向一致,表明粒子正从个体“极值”(或全局“极值”)的同侧向其逼近,需加大下代粒子的速度,以加快算法的收敛速度,其自适应调整公式为式(5)、(6)和(8)。如果 $(P_{id}^{(k)} - x_{id}^{(k)}) \times (P_{id}^{(k-1)} - x_{id}^{(k-1)}) < 0$ 或 $(P_{gd}^{(k)} - x_{id}^{(k)}) \times (P_{gd}^{(k-1)} - x_{id}^{(k-1)}) < 0$,说明 i 粒子前后两次飞行方向不一致,表明粒子正在个体“极值”(或全局“极值”)附近徘徊,需减少下代粒子的速度,以避免算法在“极值”点附近徘徊过久,提高搜索精度,其自适应调整式为(5)、(6)和(7)。如果粒子前后两次迭代有一次与个体“极值”(或全局“极值”)重合,则不改变下代粒子的速度。其数学表达式为:

$$\begin{cases} \Delta_{1id}^{(k)} = a^{(k)} \Delta_{1id}^{(k-1)}, (P_{id}^{(k)} - x_{id}^{(k)}) \times (P_{id}^{(k-1)} - x_{id}^{(k-1)}) > 0 \\ \Delta_{1id}^{(k)} = b^{(k)} \Delta_{1id}^{(k-1)}, (P_{id}^{(k)} - x_{id}^{(k)}) \times (P_{id}^{(k-1)} - x_{id}^{(k-1)}) < 0 \\ \Delta_{1id}^{(k)} = \Delta_{1id}^{(k-1)}, \text{其他} \end{cases} \quad (5)$$

$$\begin{cases} \Delta_{2id}^{(k)} = a^{(k)} \Delta_{2id}^{(k-1)}, (P_{gd}^{(k)} - x_{id}^{(k)}) \times (P_{gd}^{(k-1)} - x_{id}^{(k-1)}) > 0 \\ \Delta_{2id}^{(k)} = b^{(k)} \Delta_{2id}^{(k-1)}, (P_{gd}^{(k)} - x_{id}^{(k)}) \times (P_{gd}^{(k-1)} - x_{id}^{(k-1)}) < 0 \\ \Delta_{2id}^{(k)} = \Delta_{2id}^{(k-1)}, \text{其他} \end{cases} \quad (6)$$

第 k 代 i 粒子自适应递减因子和第 k 代 i 粒子自适应递增因子计算表达式为:

$$\begin{cases} b^{(k)} = \frac{(v_{id}^{(k-1)} - \min v_{id})}{(\max v_{id} - \min v_{id})}, \\ (P_{gd}^{(k)} - x_{id}^{(k)}) \times (P_{gd}^{(k-1)} - x_{id}^{(k-1)}) < 0 \\ b^{(k)} = \frac{v_{id}^{(k-1)}}{(\max v_{id} + \min v_{id})}, \\ (P_{id}^{(k)} - x_{id}^{(k)}) \times (P_{id}^{(k-1)} - x_{id}^{(k-1)}) < 0 \\ a^{(k)} = b^{(k)} + 1, (P_{gd}^{(k)} - x_{id}^{(k)}) \times (P_{gd}^{(k-1)} - x_{id}^{(k-1)}) > 0 \\ a^{(k)} = b^{(k)} + 1, (P_{id}^{(k)} - x_{id}^{(k)}) \times (P_{id}^{(k-1)} - x_{id}^{(k-1)}) > 0 \end{cases} \quad (7)$$

其中: $b^{(k)}$ 和 $a^{(k)}$ 分别为 i 粒子第 k 代自适应递减因子和第 k 代自适应递增因子; $V_{id}^{(k-1)}$ 为 i 粒子第 $k-1$ 代第 d 维速度分量; $\min V_{id}$ 和 $\max V_{id}$ 分别为 i 粒子前 $k-1$ 代第 d 维速度分量的最小值和最大值; $\Delta_{1id}^{(k)} = |P_{id}^{(k)} - x_{id}^{(k)}|$ 和 $\Delta_{2id}^{(k)} = |P_{gd}^{(k)} - x_{id}^{(k)}|$ 。

研究表明^[6-7],惯性因子 ω 对算法的优化性能有很大的影响,较大的 ω 值有利于提高算法的收敛速度,而 ω 较小时则

有利于提高算法的收敛精度。据此提出一种 ω 自适应调整的策略,即随着迭代的进行,逐步减少 ω 值。计算式如下:

$$\omega = k_1 \cdot S + k_2 \cdot e + \omega_0 \quad (9)$$

$$S = \left| \frac{(\max\{F_{gd}^{(k)}, F_{gd}^{(k-1)}\} - \min\{F_{gd}^{(k)}, F_{gd}^{(k-1)}\})}{f_1(k)} \right| \quad (10)$$

$$e = \frac{1}{N \times \sum_{i=1}^N \left| \frac{(F_i^{(k)} - F^{(avg)})}{f_2(k)} \right|^2} \quad (11)$$

$$\Delta F_i = |F_{gd}^{(k)} - F_{gd}^{(k-1)}| \quad (12)$$

其中: ω_0 为惯性权值的初始值; S 为粒子群进化速度; e 为群体平均适应度方差; k_1 、 k_2 为速度惯性因子和方差惯性因子; $F_{gd}^{(k)}$ 为当前迭代的全局最优解 $P_{gd}^{(k)}$ 对应的适应度; $F_{gd}^{(k-1)}$ 为上一次迭代的全局最优 $P_{gd}^{(k-1)}$ 对应的适应度; $f_1(k)$ 为归一化函数; $f_1(k) = \max\{\Delta F_1, \Delta F_2, \dots, \Delta F_k\}$; N 表示群体大小; $F_i^{(k)}$ 表示当前第 i 个粒子的适应度; $F^{(avg)}$ 表示当前粒子群的平均适应度; $f_2(k)$ 是归一化因子, $f_2(k) = \max\{|F_1^{(k)} - F^{(avg)}|, |F_2^{(k)} - F^{(avg)}|, \dots, |F_N^{(k)} - F^{(avg)}|\}$ 。

1.3 自适应变系数粒子群优化算法性能分析

自适应变系数粒子群算法是在传统粒子群算法的基础上引入了自适应递减因子 $b^{(k)}$ 和自适应递增因子 $a^{(k)}$ 及惯性因子 ω 自适应调整的策略,使算法在解搜索空间中的遍历性得到改善,因而更有可能获得全局最优,同时又可以实现局部搜索,进而提高收敛速度和提高算法精度。

2 基于 AVCP SO-RBF 神经网络混合算法

2.1 非线性的 RBF 网络结构与算法

径向基函数网络是一种三层前向神经网络, n - m - p 结构的 RBF 网络由输入层、隐含层和输出层组成,其中 n 、 m 和 p 分别为网络的输入节点、隐含层节点数和输出节点。假定第 i 个节点的激活函数为 $R_i(x)$,其形式采用最常用的高斯型径向基函数:

$$R_i(x) = \exp\left(-\frac{\|x - C_i\|^2}{2\sigma_i^2}\right); i = 1, 2, \dots, m \quad (13)$$

其中: $x \in R^n$ 为神经网络输入, C_i 为径向基函数的中心, σ_i 为基宽参数, $\| \cdot \|_n$ 为输入 x 与 C_i 之间的距离。用四元组 $\Theta = (c_i, \sigma_i, m_c, w_i)$ 表示隐节点 i 的所有参数,其中 w_i 为输出层权重。于是含有 m 个隐节点的 RBF 网络的所有参数表示为 $\xi(m) = \{\Theta_1, \Theta_2, \dots, \Theta_m\}$ 。此时的 RBF 网络可由 $\xi(m)$ 唯一标识。由此得到 RBF 网络所实现的输入输出非线性函数映射关系为:

$$y_k = f(x, \xi(m)) = WR = \sum_{i=1}^m \omega_{ik} R_i(x) \quad (14)$$

其中: W 是隐含层与输出层间的网络权向量。

RBF 神经网络学习的核心内容,是确定一组最优的网络参数 $\xi^*(m)$,使得:

$$y_i = f(x_i, \xi^*(m)) \quad (15)$$

其训练算法可表示为:对训练样本 $D_n = \{x_i, y_i\}, i = 1, 2, \dots, n$,寻找 $\Theta = (c_i, \sigma_i, m_c, w_i)$ 使函数(16)最小化。

$$\min E_{\text{RBF}} = \frac{1}{n} \sum_{i=1}^n (y_i - f(x, \xi(m)))^2 \quad (16)$$

因此,在应用 RBF 网络时,首要确定参数:中心矢量 C_i 、基宽向量 σ_i 、网络权重 w_i 。

2.2 AVCP SO-RBF 神经网络混合算法分析

从数学角度看,RBF 算法本质上是以误差平方和为目标

函数,用梯度法求其最小值的算法,而 AVCP SO 本质上属于随机寻优过程,不存在局部收敛问题;另外在许多优化问题的求解过程中,很难借助数值方程求导的方式选择优化方向,因而使经典的神经网络算法显的乏力,而粒子群算法最为广泛的应用领域是优化。

因此,可考虑在对 RBF 神经网络训练时,采用二者相结合的方法,应用 AVCP SO 对 RBF 神经网络参数($n, c_i, \sigma_i, m_c, w_i$)优化,再用 RBF 网络算法对所得网络参数进一步精确优化。这样建立的 RBF 网络模型稳定、全局收敛速度快,同时有很强的记忆能力和推广能力。

2.3 AVCP SO-RBF 混合优化算法的实现

自适应变系数粒子群—RBF 神经网络模型(AVCP SO-RBF)算法如下:AVCP SO 的粒子编码后,用 AVCP SO 算法反复优化 RBF 网络模型的参数组合($n, c_i, \sigma_i, m_c, w_i$),直至解的适应度不再有意义的增加为止,即解群质量趋于稳定。此时解码得到的参数组合已较接近符合应用需要的最佳组合。在此基础上,解码后,再用 RBF 算法对所得网络参数进一步精确优化,直至搜索到最优网络参数,此时即可得到精确的最优参数组合,也就确定了最佳网络结构。

选取 N 个粒子构成粒子群,其中每个粒子都是 D 维向量。粒子群的位置向量 x 代表 RBF 神经网络的中心矢量 C_i 、基宽向量 σ_i 、网络输出权值 w_i ,粒子初始群体随机产生后,以后各代粒子的位置根据飞行速度的变化而变化。用下式衡量每个粒子的适应值:

$$fit = \frac{1}{n} \sum_{i=1}^n \sum_{k=1}^p (y_i - y_k)^2 \quad (17)$$

其中: y_i 是第 i 个粒子的目标输出, y_k 是第 i 个粒子在 RBF 网络中实际输出, p 是输出节点数, n 是训练集样本数。

2.4 AVCP SO-RBF 混合优化算法步骤

步骤 1 初始化。给定网络的 N 组输入、输出训练样本集;确定算法的参数:根据神经网络的输入输出样本集确定网络的拓扑结构。

步骤 2 确定 RBF 网络结构后,进行 AVCP SO 的编码,并对粒子群初始化。按式(3)、(4)初始化粒子的位置 x_{id} 及速度 v_{id} ,确定粒子个数 N 、惯性因子 ω 的初值、最大允许迭代步数 k_{\max} 、加速系数 c_1 和 c_2 、初始化 P_{id} 和 P_{gd} 。

步骤 3 评价。译码每一粒子个体串,得到每个粒子对应的 RBF 网络参数,计算 N 组输入样本在此 RBF 下 N 组的输出,将其和输出样本带入以均方差作为 RBF 网络逼近的性能指标适应度函数(fit),即式(17),计算出每个粒子的个体极值(适应度值),即得到该粒子的 P_{id} 。

步骤 4 评价各粒子的目标函数,选出个体极值最好的作为全局极值,即得出粒子群的 P_{gd} ;记录该最好值的粒子序号,该粒子对应的极值就是下次迭代中 RBF 网络的最优参数。

步骤 5 判断 P_{gd} 是否满足 AVCP SO 结束条件。满足就退出 AVCP SO 寻优,转入 RBF 局部优化即步骤 7。

步骤 6 对每一粒子进行速度和位置更新。即更新极值。计算出每一粒子的适应度值 fit_i ,若优于该粒子当前的个体极值,则将 P_{id} 设置为该粒子的位置,且更新个体极值。若所有粒子的个体极值中最好的优于当前全局极值,则将 P_{gd} 设置为该粒子的位置,记录该粒子的序号,且更新全局极值;根据式(9)更新惯性权重;根据式(3)、(4)更新位置和速度;再转入步骤 3。

步骤7 译码 P_{gd} 所对应的那个粒子的个体串,以此进化值为 RBF 网络的初值,进行局部优化。

步骤8 检验。对局部优化后的参数进行编码,判断是否满足 AVCPSO 结束条件,通常采用结束条件为 P_{gd} 达到一定的设定值,迭代达到最大次数或最小误差要求时,停止迭代,若偏差超出设定目标,再返回步骤6,进行 AVCPSO 全局寻优。

3 AVCPSO-RBF 在短期电力负荷预测的应用

3.1 仿真计算

根据以上自适应变系数粒子群—RBF 网络混合算法模型,采用贵州电网 2007 年 7 月到 2008 年 8 月的负荷数据、天气数据和气象数据,以 Matlab 为工具对贵州电网一天 24 小时进行负荷预测。

采用 3 层 RBF 神经网络模型:输入层节点数、隐含层节点数未定由 AVCPSO 算法优化确定,输出层 1 个,代表预测日一个时段的预测负荷值。

在实验中,AVCPSO-RBF 的初始参数分别为: $\omega_0 = 0.6$, $c_1 = 2.1$ 和 $c_2 = 2.3$, k_i 取值为 $[0.1, +1.1]$, ω_i 取值为 $[-1, +1]$, r_1, r_2 取值为 $[0, 1]$, 粒子群数 $Size = 80$, 粒子速度取值为 $[0, +10]$ 。

采用 AVCPSO-RBF 程序、粒子群—RBF 神经网络混合算法(PSO-RBF)和 RBF 神经网络算法程序对贵州电网 2008 年 8 月 8 日和 9 日的各小时负荷进行预测。表 1、图 1 给出了预测结果。

从表 1 和图 1 中可以看出,AVCPSO-RBF 网络混合算法预测的负荷与实际负荷相当接近。

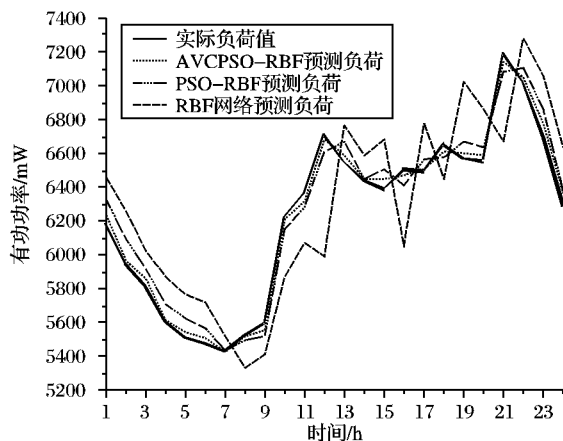


图 1 2008 年 8 月 9 日 24h 实际和预测的电力负荷

表 1 2008 年 8 月 8 日预测结果比较

模型	平均实际负荷/mW	平均预测值/mW	平均误差/%	均方误差	合格率/%
AVCPSO-RBF 方法	6241.9	6140.2	1.63	1.86	100.0
PSO-RBF 方法	6241.9	5970.4	4.35	5.44	83.3
RBF 网络方法	6241.9	5693.9	8.78	10.37	66.7

3.2 仿真效果分析

为评价 AVCPSO-RBF 神经网络模型的仿真效果,本文采用传统的 BP 神经网络模型、RBF 网络模型和 PSO-RBF 神经网络模型分别进行模拟,模型训练结果见表 2。在相同控制精度条件下,BP、RBF、PSO-RBF 和 AVCPSO-RBF 模型的迭代次数分别为 4500、900、300 和 120,平均绝对值相对误差分别

为 11.692%、8.791%、4.339% 和 1.631%。

表 2 网络模型训练性能对比

模型	控制精度	迭代次数	平均绝对值相对误差/%	最大绝对值相对误差/%	训练时间/s
BP	0.005	4500	11.692	15.550	181
RBF	0.005	900	8.791	15.420	122
PSO-RBF	0.005	300	4.339	9.590	98
AVCPSO-RBF	0.005	120	1.631	2.536	78

从表 2 可以看出,自适应变系数粒子群算法训练 RBF 神经网络的误差、训练时间和迭代次数都明显优于传统算法,即 AVCPSO-RBF 神经网络算法比 BP 网络、RBF 网络和 PSO-RBF 模型算法有着更快的收敛速度和更高的收敛精度。

图 2 为 AVCPSO-RBF 神经网络模型与 PSO-RBF 神经网络模型的性能比较。从图中看出,当迭代次数为 100, PSO-RBF 网络适应度函数就达到稳定,函数值相应的比较高。而 AVCPSO-RBF 神经网络迭代次数达到 130 才达到稳定。将粒子群的适应度函数设定为神经网络的训练误差,适应度函数越大,输出误差就越大,可见 AVCPSO-RBF 网络模型的精度远高于 PSO-RBF 神经网络模型的精度。AVCPSO-RBF 网络模型更实用。

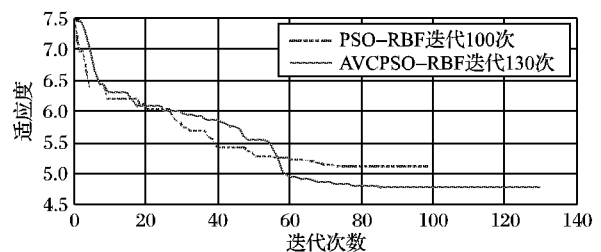


图 2 适应度函数值比较

表 3 给出了某日应用自适应变系数粒子群算法—RBF 神经网络模型(AVCPSO-RBF)和粒子群—RBF 神经网络模型(PSO-RBF)预测的某日各时段电力负荷的平均绝对值相对误差 E_{MAPE} ,其中 E_{MAPE} 按式(18)计算。

$$E_{MAPE} = \frac{1}{m} \sum_{j=1}^m \frac{|\hat{p}_{ij} - p_{ij}|}{\hat{p}_{ij}} \times 100\% \quad (18)$$

其中: p_{ij} 、 \hat{p}_{ij} 为第 j 天第 i 时段的实际电力负荷和预测电力负荷; \hat{p}_{ij} 为第 j 天第 i 时段的实际电力负荷的均值; $i = 1, 2, \dots, 24$ 为时段数; $j = 1, 2, \dots, 7$ 为预测的天数。

表 3 某日两种模型各时段的预测误差 E_{MAPE} %

时段 i	AVCPSO-RBF	PSO-RBF	时段 i	AVCPSO-RBF	PSO-RBF
1	1.87	6.9633	13	1.13	3.0266
2	1.24	6.9511	14	1.38	1.9441
3	2.35	4.5918	15	1.23	1.1527
4	2.00	1.9231	16	1.41	3.1088
5	1.18	9.1735	17	1.57	1.7923
6	2.52	9.5782	18	1.30	1.8697
7	2.01	6.1337	19	1.52	3.9275
8	1.69	3.6349	20	2.54	4.1302
9	1.53	5.0392	21	1.65	6.2693
10	2.19	3.8503	22	1.60	2.2829
11	1.16	1.3219	23	1.10	8.4152
12	1.11	1.7372	24	1.82	5.2724

从表 3 中可以看出自适应变系数粒子群—RBF 神经网络模型所预测的贵州电网电力负荷大部分时段的 E_{MAPE} 都保持

在 2% 以内,平均 E_{MAPE} 为 1.63%,没有出现误差超过 3% 的时段,而粒子群—RBF 神经网络模型则出现了 7 个超过 6% 的时段,甚至出现了 2 个超过 9% 的时段,平均 E_{MAPE} 为 4.34%。从表 2~3 可知采用自适应变系数粒子群算法的 RBF 神经网络方法的预测精度明显好于传统 BP 神经网络方法、传统 RBF 神经网络方法、传统粒子群算法的 RBF 神经网络方法。

根据 AVCPSO-RBF 网络模型,预测出 2008 年 8 月 26 日的贵州电力负荷,将其与混沌理论神经网络模型所得的结果进行比较。

表 4 两种模型的误差值对比

误差	混沌神经	AVCPSO-RBF	混沌神经	AVCPSO-RBF
	网络模型	模型	网络模型	模型
	最大值/%	最大值/%	平均值/%	平均值/%
相对误差	21.557	2.539	7.043	1.633
中值相对误差	13.825	2.434	5.022	1.557

从表 4 可见 AVCPSO-RBF 网络模型的平均相对误差、平均中值相对误差等各项误差分析指标都优于混沌模型。

从图 3 中能够清楚地看出本文采用的自适应变系数粒子群—RBF 神经网络在进行次日电力负荷预测中的优越性。

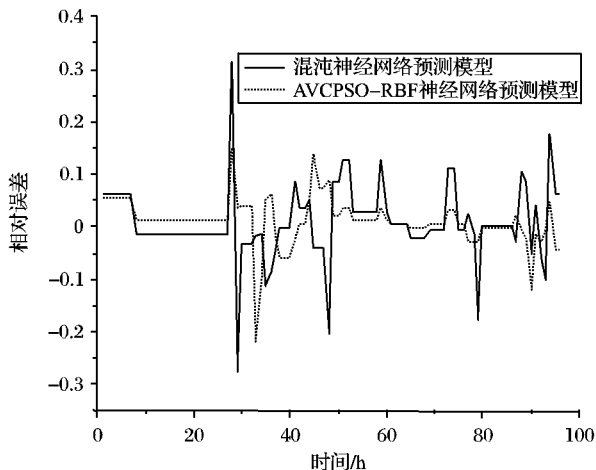


图 3 两种模型的预测相对误差曲线对比

总之,自适应变系数粒子群—RBF 神经网络模型相对于基于混沌理论的神经网络模型,在减小迭代次数、提高预测精度方面有了很大的改进。由于改进的粒子群算法代替了 RBF

网络的初始寻优,网络仅在已接近最优解的基础上进行参数寻优,从而有效提高了网络寻优速度和精度。

4 结语

本文分析了传统的粒子群算法及缺点,提出了基于自适应变系数粒子群—RBF 神经网络模型的短期电力负荷预测方法。应用自适应变系数粒子群算法反复优化 RBF 网络模型的参数组合 (C_i, σ_i, w_i) , 得到最优粒子对应的 RBF 网络参数组合,再用 RBF 网络算法对所得网络参数进一步精确优化,得到精确的最优参数组合,实现 RBF 网络参数优化。利用电网历史数据进行短期电力负荷预测,算例表明:该模型成功地解决了贵州电网短期电力负荷预测精度不高的难题;该方法具有输出稳定性好、收敛性快、预测精度高优点,各日预测电力负荷的百分比误差可控制在 1.7% 以内,并克服了 RBF 网络和传统的粒子群优化法的缺点。

在减小迭代次数、提高预测精度方面自适应变系数粒子群—RBF 神经网络模型优于混沌理论的神经网络模型和粒子群—RBF 神经网络模型及 RBF 网络模型。该方法改善了 RBF 网络泛化能力,在电力系统短期负荷预测中有良好的应用效果,为电力系统短期负荷预测提供了一种科学、有效的方法。

参考文献:

- [1] 姚建刚,章建,银车来. 电力市场运营及其软件开发[M]. 北京: 中国电力出版社, 2002.
- [2] 康重庆,周安石,王鹏,等. 短期负荷预测中实时气象因素的影响分析及其处理策略[J]. 电网技术, 2006, 30(7): 5-10.
- [3] 雷绍兰,孙才新,周源,等. 基于径向基神经网络和自适应神经模糊系统的电力短期负荷预测方法[J]. 中国电机工程学报, 2005, 25(22): 78-82.
- [4] SENTHIL BRUMMAGEM M, RAO M V C, CHANDRAMOHAN A. A new and improved version of particle swarm optimization algorithm with global-local best parameters [J]. Knowledge and Information Systems, 2008, 16(3): 15-26.
- [5] 李勇刚,桂卫华,阳春华,等. 一种弹性粒子群优化算法[J]. 控制与决策, 2008, 23(1): 95-98.
- [6] 张选平,杜玉平,秦国强,等. 一种动态改变惯性权的自适应粒子群算法[J]. 西安交通大学学报, 2005, 39(10): 1039-1042.
- [7] 师彪,李郁侠,于新花,等. 改进粒子群—BP 神经网络模型的短期电力负荷预测[J]. 计算机应用, 2009, 29(4): 1036-1039.
- [8] GE G N, FUJI-HARA R, MIAO Y. Further combinatorial constructions for optimal frequency hopping sequences [J]. Journal of Combinatorial Theory Series A, 2006, 113(8): 1699-1718.
- [9] CHU W S, COLBOURN C J. Optimal frequency-hopping sequences via cyclotomy [J]. IEEE Transactions on Information Theory, 2005, 51(3): 1139-1141.
- [10] DING C S, MARKO J M. Algebraic constructions of optimal frequency-hopping sequences [J]. IEEE Transactions on Information Theory, 2007, 53(7): 2606-2610.
- [11] PEND DAI-YUAN, FAN PING-ZHI. Lower bounds on the Hamming auto- and cross correlations of frequency-hopping sequences [J]. IEEE Transactions on Information Theory, 2004, 50(9): 2149-2154.
- [12] CARLET C, DING C S. Highly nonlinear mappings [J]. Journal of Complexity, 2004, 20(2/3): 205-244.

(上接第 2438 页)

参考文献:

- [1] KUMAR P V. Frequency-hopping code sequence designs having large linear span [J]. IEEE Transactions on Information Theory, 1988, 34(1): 146-151.
- [2] 梅文华,杨义先. 基于 $GF(p^k)$ 上 m 序列的最佳跳频序列族[J]. 通信学报, 1996, 17(2): 12-15.
- [3] 梅文华,杨义先. 基于 GMW 序列构造最佳跳频序列族[J]. 通信学报, 1997, 18(11): 20-24.
- [4] UDAYA P, SIDDIQI M N. Optimal large linear complexity frequency hopping patterns derived from polynomial residue class rings [J]. IEEE Transactions on Information Theory, 1998, 44(4): 1492-1503.
- [5] FUJI-HARA R, MIAO Y, MISHIMA M. Optimal frequency hopping sequences: A combinatorial approach [J]. IEEE Transactions on Information Theory, 2004, 50(10): 2408-2420.