

基于人工智能技术的火电厂燃煤锅炉智能燃烧优化研究及应用

牛培峰¹, 马云鹏², 张欣欣¹, 胡晓宾¹

(1. 燕山大学电气工程学院, 河北 秦皇岛 066004;

2. 天津商业大学信息工程学院, 天津 300134)

摘要: 为了降低火电厂燃煤锅炉的氮氧化合物排放浓度和锅炉煤耗, 提出了样本增量量子神经网络和改进型量子蜂群算法。样本增量量子神经网络可以动态建立燃煤锅炉的氮氧化合物排放浓度和锅炉煤耗的综合模型, 并且可实现模型滚动优化。基于建立的综合模型, 通过应用改进型量子蜂群算法优化一二次风量、燃煤量和各二次风门开度来实现锅炉燃烧优化。基于上述两种方法, 开发了一套燃煤锅炉智能燃烧优化软件, 并应用于某热电厂 330 MW 锅炉上, 测试结果表明, 氮氧化合物排放浓度和锅炉煤耗均有不同程度的降低, 说明建模方法和优化算法是有效的。

关键字: 氮氧化合物排放; 锅炉; 神经网络; 量子计算; 蜂群算法

中图分类号: TP323

文献标识符: A

doi: 10.11959/j.issn.2096-6652.201925

Research and application on combustion optimization of coal-fired boiler in thermal power plant based on artificial intelligence technology

NIU Peifeng¹, MA Yunpeng², ZHANG Xinxin¹, HU Xiaobin¹

1. School of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China

2. School of Information Engineering, Tianjin University of Commerce, Tianjin 300134, China

Abstract: In order to reduce the NO_x emission concentration and coal consumption of coal-fired boilers in thermal power plants, the sample increment quantum neural network and an improved quantum bee colony algorithm were proposed. The quantum sample incremental feed-forward neural network can dynamically establish a comprehensive model of Nitrogen and Oxygen emission concentration and boiler coal consumption of coal-fired boiler, and can realize rolling optimization of the model. Based on the established comprehensive model, the optimization of the primary and secondary air volume and coal and the opening degree of each secondary air valve were realized by using the improved quantum bee colony algorithm. Based on the above two methods, a set of intelligent combustion optimization software for coal-fired boiler was developed and applied to the 330 MW boiler of a thermal power plant. The test results show that the Nitrogen and Oxygen emission concentration and the coal consumption of the boiler have been reduced in varying degrees. It is shown that the modeling method and the optimization algorithm are effective.

Key words: NO_x emission, boiler, neural network, quantum computation, bee colony algorithm

收稿日期: 2019-04-23; 修回日期: 2019-06-03

通信作者: 马云鹏, 18232362368@163.com

基金项目: 国家自然科学基金资助项目 (No.61573306)

Foundation Item: The National Natural Science Foundation of China (No.61573306)

1 引言

伴随着我国经济的高速发展,能源的使用量越来越高,其中电能的使用量与日俱增。据中国电力企业联合会行业发展与环境资源部统计,2017年1~10月我国总的用电量为52 018亿千瓦时,同比增长6.7%。据统计,我国是世界上拥有火电燃煤机组最多的国家之一,电力行业煤炭消费量(动力煤)占煤炭总消费量的50%以上。因此,对火力发电提出了两大亟须解决的问题:一是提高机组运行效率、降低机组煤耗;二是在燃煤过程中有效降低有害气体排放量^[1]。

(1) 提高机组运行效率和降低机组煤耗的必要性^[1]

我国的火力发电企业面临“厂网分开、竞价上网”的电力市场竞争现状,而且机组的运行效率与发达国家相比仍存在不小的差距^[2]。因此,在燃煤机组安全运行的条件下,提高燃煤发电机组的运行效率,同时降低机组煤耗,最大限度地降低火电企业的生产成本,是提高市场竞争力的重要手段。

(2) 在燃煤过程中有效降低有害气体排放量的必要性^[1]

众所周知,火力发电企业使用大量的煤炭将会导致严重的大气环境污染问题。煤炭燃烧过程会产生大量的有害气体,如 CO_2 、 NO_x 和 SO_2 等,它们与酸雨、温室效应和光化学烟雾的形成以及地球臭氧层的破坏有直接关系^[3],且会导致雾霾严重、PM2.5超标等,给人们的生活和健康造成威胁。2018年“两会”期间,政府工作报告将 SO_2 和 NO_x 排放量下降3%作为当年的工作重点。因此,在保证锅炉机组安全运行的前提下,如何有效地降低污染物排放是火电厂必须要解决的问题^[1]。

针对上述两个问题,文献[4]利用改进的最小资源分配神经网络建立锅炉热效率和 NO_x 的综合模型,并采用基于实数编码的遗传算法优化锅炉燃烧系统,达到提高锅炉热效率和降低 NO_x 排放浓度的综合目标。文献[5]采用量子遗传算法优化基于最小二乘支持向量机的锅炉热效率和 NO_x 排放浓度模型,提高了模型精度和泛化能力。在此模型的基础上,应用博弈差分算法实现了锅炉多目标燃烧优化,改善了运行工况。文献[6]为提高CFBB(circulating fluidized bed boiler,循环流化床锅炉)的热效率和降低污染气体的排

放浓度,应用自适应模糊推理辨识方法建立热效率、 NO_x 和 SO_2 排放浓度模型,并使用果蝇优化算法对锅炉的运行工况进行优化,寻找锅炉运行的最佳工况参数,保证其高效率、低排放运行。文献[7]提出了一种基于大数据驱动案例匹配的锅炉燃烧优化算法,利用数据挖掘技术对SIS(safety instrumented system,安全仪表)系统中的数据进行分析并建立燃烧案例库。在线优化时,可将DCS(distributed control system,分布式控制系统)的实时运行数据与案例库进行案例匹配,找到适合当前工况的最佳运行参数,实现锅炉燃烧在线优化。

然而,迄今为止,能够满足燃煤机组大范围变负荷的动态优化技术和方法还不多见。为解决此类问题,本文提出采用样本增量量子神经网络和一种改进型量子蜂群算法实现锅炉燃烧过程的智能燃烧优化。样本增量量子神经网络可以动态建立燃煤锅炉的 NO_x 排放浓度和锅炉煤耗的综合优化模型,并且可实现模型参数的滚动优化,以保证模型的辨识能力和泛化能力,实现燃煤机组大范围变负荷的动态优化。基于建立的优化模型,应用改进型量子蜂群算法优化一二次风量、燃煤量和各二次风门开度来实现锅炉燃烧优化。基于上述两种方法,开发了一套燃煤锅炉智能燃烧优化软件,并应用于某热电厂330 MW锅炉上,应用测试结果表明: NO_x 的排放浓度和锅炉煤耗均有不同程度的降低,从而验证了建模方法和优化算法的有效性。

2 样本增量量子神经网络

本文提出了一种具有在线学习能力的样本增量量子神经网络(sample increment quantum neural network, SIQNN),其输入权值为量子位权值,即根据量子计算规则设置输入权值;隐层神经元为量子神经元;输出神经元可以直接获取输入神经元的信息;网络参数可以根据样本信息调节。SIQNN的模型结构和隐藏层量子神经元分别如图1和图2所示。本文对量子计算规则不做详细叙述,其详细计算过程可参考文献[8]。

如图1所示,SIQNN有 n 个输入神经元, m 个隐层神经元和 l 个输出神经元。假设有 N 个随机样本 $(\mathbf{x}_i, \mathbf{y}_i)$, $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$ 为第 i 组输入样本数据, $\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{il}]^T \in R^l$ 为第 i 组相应的输

出数据。 $\{|\omega^{in}\rangle\}_{m \times n}$ 为输入层与隐层之间的量子权值， $\mathbf{b}=[b_1, b_2, \dots, b_m]$ 为隐层的阈值向量， $\mathbf{c}=[c_1, c_2, \dots, c_l]$ 为输出层阈值向量， $\beta_{l \times m}$ 表示输出权值矩阵， ω^{oi} 是输入层与输出层之间的权值矩阵。图2为一个隐层神经元的示意，参考文献[9]中隐层神经元的构建思想包括加权、聚合、活化和激励四部分：加权体现在量子输入权值上，增强输入数据的影响；聚合是将不同的输入神经元的信息汇聚到一起；活化是加强量子神经元的活性，其中 $|\phi\rangle$ 为量子活化值；激励体现在隐层的激励函数 $f(\cdot)$ 中。量子神经元的数学描述表达如下。

$$t_j = f(\mathbf{x}^T |\omega\rangle, |\phi\rangle - b_j) = f\left(\sum_{i=1}^n x_i \langle \omega_{ji}, \varphi_{ji} \rangle - b_j\right) \quad (1)$$

式(1)中， $|\omega_j\rangle = (|\omega_{j1}\rangle, |\omega_{j2}\rangle, \dots, |\omega_{jn}\rangle)^T$ 为量子权值，记作 $|\omega_{jn}\rangle = (\cos \theta_{jn}, \sin \theta_{jn})^T$ ， $\theta_{ij} \in [0, 2\pi]$ ； $|\phi\rangle = (|\phi_{j1}\rangle, |\phi_{j2}\rangle, \dots, |\phi_{jn}\rangle)$ 为活化权值，记为 $|\phi_{jn}\rangle = (\cos \zeta_{jn}, \sin \zeta_{jn})^T$ ， $\zeta_{ij} \in [0, \pi]$ 。根据内积计算规则，式(1)可以转化为如下形式。

$$t_j = f\left(\sum_{i=1}^n x_i \cos(\theta_{ji} - \zeta_{ji}) - b_j\right) \quad (2)$$

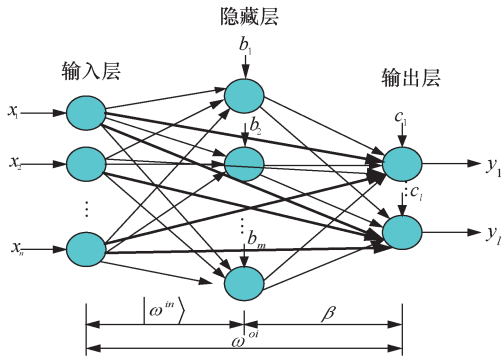


图1 样本增量量子神经网络结构

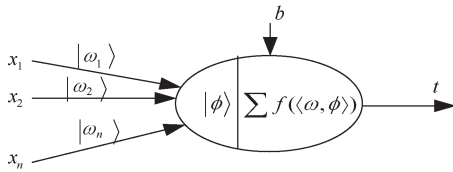


图2 隐层神经元

2.1 算法逼近问题描述

根据图1，SIQNN的输出神经元的数学模型可以描述如下。

$$\begin{cases} y_{j1} = g\left(\sum_{r=1}^n \omega_{1r}^{oi} x_{jr} + c_1 + \sum_{k=1}^m \beta_{1k} t_k\right) \\ y_{j2} = g\left(\sum_{r=1}^n \omega_{2r}^{oi} x_{jr} + c_2 + \sum_{k=1}^m \beta_{2k} t_k\right), j=1, 2, \dots, N \\ \vdots \\ y_{jl} = g\left(\sum_{r=1}^n \omega_{lr}^{oi} x_{jr} + c_l + \sum_{k=1}^m \beta_{lk} t_k\right) \end{cases} \quad (3)$$

式(3)可以转化为如下形式。

$$\mathbf{y}_j = g\left(\omega^{oi} \mathbf{x}_j + \mathbf{c} + \sum_{k=1}^m \beta_k t_k\right), j=1, 2, \dots, N \quad (4)$$

式(4)可以转化为如下矩阵形式。

$$\mathbf{Y} = g(\omega^{oi} \mathbf{X} + \beta \mathbf{T} + \mathbf{c}) = g\left(\begin{bmatrix} \omega^{oi} & \beta & \mathbf{c} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{T} \\ \mathbf{I} \end{bmatrix}\right) = g\left(\mathbf{W} \begin{bmatrix} \mathbf{X} \\ \mathbf{T} \\ \mathbf{I} \end{bmatrix}\right) \quad (5)$$

式(5)中的 $\mathbf{T}$ 为隐层输出矩阵， $\mathbf{W}$ 为输出复合权值矩阵。

根据以上推导过程，SIQNN可以被当作一个线性系统，因此输出复合权值矩阵可由式(6)解析求得。

$$\left\| g\left(\begin{bmatrix} \hat{\mathbf{W}} \\ \mathbf{T} \\ \mathbf{I} \end{bmatrix}\right) - \mathbf{Y} \right\| = \min_{\hat{\mathbf{W}}} \left\| g\left(\begin{bmatrix} \mathbf{W} \\ \mathbf{T} \\ \mathbf{I} \end{bmatrix}\right) - \mathbf{Y} \right\| \quad (6)$$

假如激励函数 $g(\cdot)$ 是可逆的，则式(6)可以转化为下式。

$$\left\| \begin{bmatrix} \hat{\mathbf{W}} \\ \mathbf{T} \\ \mathbf{I} \end{bmatrix} - g^{-1}(\mathbf{Y}) \right\| = \min \left\| \begin{bmatrix} \mathbf{W} \\ \mathbf{T} \\ \mathbf{I} \end{bmatrix} - g^{-1}(\mathbf{Y}) \right\| \quad (7)$$

根据MP广义逆计算规则，复合矩阵 $\hat{\mathbf{W}}$ 的解表示如下。

$$\hat{\mathbf{W}} = g^{-1}(\mathbf{Y}) \begin{bmatrix} \mathbf{X} \\ \mathbf{T} \\ \mathbf{I} \end{bmatrix}^+ = g^{-1}(\mathbf{Y}) \mathbf{H}^+ \quad (8)$$

其中， $\mathbf{H} = [\mathbf{X} \ \mathbf{T} \ \mathbf{I}]^T = [\mathbf{X}^T \ \mathbf{T}^T \ \mathbf{I}^T]$ 。

根据复合矩阵，输入层与输出层之间的权值矩阵、输出权值矩阵和输出层阈值向量分别表示如下。

$$\begin{cases} \omega^{oi} = \hat{\mathbf{W}}(1:l, 1:n) \\ \beta = \hat{\mathbf{W}}(1:l, (n+1):(n+m)) \\ \mathbf{c} = \hat{\mathbf{W}}(1:l, n+m+1) \end{cases} \quad (9)$$

2.2 初始化阶段

首先, 随机采集 L 组数据作为初始训练样本 $\{(\mathbf{x}_i, \mathbf{t}_i)\}_{i=1}^L$; 然后, 采用量子计算规则初始化输入权值和隐层阈值; 最后, 基于式 (8) 计算复合矩阵 $\tilde{\mathbf{W}}_0$, 根据式 (9) 计算输出权值和输入层与输出层之间的权阈值, 并保存初始化模型参数。

2.3 贯序学习阶段

根据样本之间的增量动态调节输入权阈值、输出权阈值以及输入层与输出层之间的权值, 保证了模型参数对输入数据的自适应性, 避免了相同数据的再学习问题和隐层输出权值的病态问题, 可提高算法的计算速度。

隐层量子神经元的输出矩阵表示如下。

$$\mathbf{T} = \begin{bmatrix} f(\omega_1 \mathbf{x}_1 - b_1) & \cdots & f(\omega_1 \mathbf{x}_N - b_1) \\ \vdots & \ddots & \vdots \\ f(\omega_m \mathbf{x}_1 - b_m) & \cdots & f(\omega_m \mathbf{x}_N - b_m) \end{bmatrix}_{m \times N} \quad (10)$$

由式 (5) 可以得到

$$\beta \mathbf{T} = \mathbf{g}^{-1}(\mathbf{Y}) - \omega^{oi} \mathbf{X} - \mathbf{c} \quad (11)$$

$$\mathbf{T} = \beta^+ (\mathbf{g}^{-1}(\mathbf{Y}) - \omega^{oi} \mathbf{X} - \mathbf{c}) \quad (12)$$

假设存在一个随机矩阵 $\mathbf{p}$ 使得输入权阈值的复合矩阵 $\tilde{\omega}_0$ 表示成如下形式。

$$\tilde{\omega}_0 = \mathbf{p} \mathbf{X}_0 \tilde{\mathbf{X}}_0^T = \mathbf{p} \mathbf{X}_0 \tilde{\mathbf{X}}_0^T (\tilde{\mathbf{X}}_0 \tilde{\mathbf{X}}_0^T)^{-1} \quad (13)$$

假设实时数据逐个输入模型中, 当新数据 $\mathbf{x}_1$ 到来时, 计算复合矩阵 $\tilde{\omega}_1$ 。

$$\begin{aligned} \tilde{\omega}_1 &= \mathbf{p} [\mathbf{X}_0 \quad \mathbf{x}_1] [\tilde{\mathbf{X}}_0 \quad \tilde{\mathbf{x}}_1]^T \left([\tilde{\mathbf{X}}_0 \quad \tilde{\mathbf{x}}_1] [\tilde{\mathbf{X}}_0 \quad \tilde{\mathbf{x}}_1]^T \right)^{-1} = \\ &= \mathbf{p} [\mathbf{X}_0 \quad \mathbf{x}_1] \begin{bmatrix} \tilde{\mathbf{X}}_0^T \\ \tilde{\mathbf{x}}_1^T \end{bmatrix} \left(\begin{bmatrix} \tilde{\mathbf{X}}_0^T \\ \tilde{\mathbf{x}}_1^T \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{X}}_0^T \\ \tilde{\mathbf{x}}_1^T \end{bmatrix} \right)^{-1} = \\ &= (\mathbf{p} \mathbf{X}_0 \tilde{\mathbf{X}}_0^T + \mathbf{p} \mathbf{x}_1 \tilde{\mathbf{x}}_1^T) (\tilde{\mathbf{X}}_0 \tilde{\mathbf{X}}_0^T + \tilde{\mathbf{x}}_1 \tilde{\mathbf{x}}_1^T)^{-1} = \\ &= [\mathbf{p} \mathbf{X}_0 \tilde{\mathbf{X}}_0^T (\tilde{\mathbf{X}}_0 \tilde{\mathbf{X}}_0^T)^{-1} (\tilde{\mathbf{X}}_0 \tilde{\mathbf{X}}_0^T) + \\ &+ \mathbf{p} \mathbf{x}_1 \tilde{\mathbf{x}}_1^T (\tilde{\mathbf{X}}_0 \tilde{\mathbf{X}}_0^T + \tilde{\mathbf{x}}_1 \tilde{\mathbf{x}}_1^T)^{-1} = \\ &= [\tilde{\omega}_0 (\tilde{\mathbf{X}}_0 \tilde{\mathbf{X}}_0^T) + \mathbf{p} \mathbf{x}_1 \tilde{\mathbf{x}}_1^T] (\tilde{\mathbf{X}}_0 \tilde{\mathbf{X}}_0^T + \tilde{\mathbf{x}}_1 \tilde{\mathbf{x}}_1^T)^{-1} \end{aligned} \quad (14)$$

假设 $\mathbf{K}_0 = \tilde{\mathbf{X}}_0 \tilde{\mathbf{X}}_0^T$, $\mathbf{K}_1 = \tilde{\mathbf{X}}_0 \tilde{\mathbf{X}}_0^T + \tilde{\mathbf{x}}_1 \tilde{\mathbf{x}}_1^T$, 于是式 (14) 可以简化为如下形式。

$$\begin{aligned} \tilde{\omega}_1 &= (\tilde{\omega}_0 \mathbf{K}_0 + \mathbf{p} \mathbf{x}_1 \tilde{\mathbf{x}}_1^T) \mathbf{K}_1^{-1} = \\ &= [\tilde{\omega}_0 (\mathbf{K}_1 - \tilde{\mathbf{x}}_1 \tilde{\mathbf{x}}_1^T) + \mathbf{p} \tilde{\mathbf{x}}_1 \tilde{\mathbf{x}}_1^T] \mathbf{K}_1^{-1} = \\ &= \tilde{\omega}_0 + (\mathbf{p} \mathbf{x}_1 - \tilde{\omega}_0 \tilde{\mathbf{x}}_1) \tilde{\mathbf{x}}_1^T \mathbf{K}_1^{-1} \end{aligned} \quad (15)$$

针对实时数据建模, 采集数据的周期可能很小

或被采集的数据在很长时间内没有大的变化, 此时没有必要更新 SIQNN 的模型参数。因此, 当新样本数据与前一时刻的数据之间存在增量时, 模型参数根据增量进行调整; 否则, 保持不变。新样本数据与旧样本数据之间的增量是指两组数据做向量的减法所得到的差值向量, 如果增量的秩为 1, 说明两组数据之间有差异; 如果增量的秩为 0, 说明两组数据相同。因此, 式 (15) 可以变形为如下的形式。

$$\begin{aligned} \tilde{\omega}_1 &= \tilde{\omega}_0 + [\mathbf{p}(\mathbf{x}_1 - \mathbf{x}_L + \mathbf{x}_L) - \tilde{\omega}_0(\tilde{\mathbf{x}}_1 - \tilde{\mathbf{x}}_L + \tilde{\mathbf{x}}_L)] \tilde{\mathbf{x}}_1^T \mathbf{K}_1^{-1} = \\ &= \tilde{\omega}_0 + (\mathbf{p} \Delta \mathbf{x}_1 + \mathbf{p} \mathbf{x}_L - \tilde{\omega}_0 \Delta \tilde{\mathbf{x}}_1 - \tilde{\omega}_0 \tilde{\mathbf{x}}_L) \tilde{\mathbf{x}}_1^T \mathbf{K}_1^{-1} = \\ &= \tilde{\omega}_0 + (\mathbf{p} \Delta \mathbf{x}_1 + \mathbf{p} \mathbf{x}_L - \tilde{\omega}_0 \Delta \tilde{\mathbf{x}}_1 - \mathbf{p} \mathbf{x}_L) \tilde{\mathbf{x}}_1^T \mathbf{K}_1^{-1} = \\ &= \tilde{\omega}_0 + (\mathbf{p} \Delta \mathbf{x}_1 - \tilde{\omega}_0 \Delta \tilde{\mathbf{x}}_1) \tilde{\mathbf{x}}_1^T \mathbf{K}_1^{-1} \end{aligned} \quad (16)$$

式 (16) 中, $\tilde{\omega}_0$ 为量子权值和隐层阈值组合成的初始复合矩阵, $\Delta \mathbf{x}_1$ 为新数据 $\mathbf{x}_1$ 与上一时刻的数据 $\mathbf{x}_L$ 之间的差值向量, 当且仅当 $\Delta \mathbf{x}_1$ 的秩等于零 (即 $\Delta \mathbf{x}_1$ 的所有元素均为 0) 时, $\tilde{\omega}_1 = \tilde{\omega}_0$ 。

根据式 (16) 的复合矩阵 $\tilde{\omega}_1$ 更新输入权值和隐层阈值, 如式 (17) 和式 (18) 所示。

$$\omega_1 = \tilde{\omega}_1(1:m, 1:n) \quad (17)$$

$$b_1 = \tilde{\omega}_1(1:m, n+1) \quad (18)$$

其中, ω_1 为更新后的输入权值矩阵, b_1 为更新后的隐层阈值矩阵。

输入权值和隐层阈值确定之后, 根据新增样本数据 $\mathbf{x}_1$ 计算相应的复合矩阵 $\mathbf{W}_1$ 。当新数据到来时, 复合矩阵 $\mathbf{W}_1$ 可以由式 (19) 计算得到。

$$\begin{aligned} &\left\| \mathbf{g} \left(\begin{bmatrix} \mathbf{X}_0 \mathbf{x}_1 \\ \mathbf{T}_0 \mathbf{T}_1 \\ \mathbf{I}_0 \mathbf{I}_1 \end{bmatrix} \right) - [\mathbf{Y}_0 \quad \mathbf{Y}_1] \right\| = \\ &\min_{\mathbf{W}_1} \left\| \mathbf{g} \left(\begin{bmatrix} \mathbf{X}_0 \mathbf{x}_1 \\ \mathbf{T}_0 \mathbf{T}_1 \\ \mathbf{I}_0 \mathbf{I}_1 \end{bmatrix} \right) - [\mathbf{Y}_0 \quad \mathbf{Y}_1] \right\| \end{aligned} \quad (19)$$

令 $\boldsymbol{\eta}_0 = [\mathbf{X}_0^T \quad \mathbf{T}_0^T \quad \mathbf{I}_0^T]^T$, $\boldsymbol{\eta}_1 = [\mathbf{x}_1^T \quad \mathbf{T}_1^T \quad \mathbf{I}_1^T]^T$, 当激励函数 $\mathbf{g}(\mathbf{x})$ 可逆时, 式 (19) 可以转化为如下形式。

$$\begin{aligned} &\left\| \hat{\mathbf{W}}_1 [\boldsymbol{\eta}_0 \quad \boldsymbol{\eta}_1] - \mathbf{g}^{-1}([\mathbf{Y}_0 \quad \mathbf{Y}_1]) \right\| = \\ &\min_{\mathbf{W}_1} \left\| \mathbf{W}_1 [\boldsymbol{\eta}_0 \quad \boldsymbol{\eta}_1] - \mathbf{g}^{-1}([\mathbf{Y}_0 \quad \mathbf{Y}_1]) \right\| \end{aligned} \quad (20)$$

根据 MP 广义逆定理, 式 (20) 的最小二乘范数解为

$$\begin{aligned} W_1 &= g^{-1}([Y_0 \ Y_1])[\eta_0 \ \eta_1]^T ([\eta_0 \ \eta_1][\eta_0 \ \eta_1]^T)^{-1} = \\ &[g^{-1}(Y_0) \ g^{-1}(Y_1)] \begin{bmatrix} \eta_0^T \\ \eta_1^T \end{bmatrix} \left(\begin{bmatrix} \eta_0 & \eta_1 \end{bmatrix} \begin{bmatrix} \eta_0^T \\ \eta_1^T \end{bmatrix} \right)^{-1} = \\ &(g^{-1}(Y_0)\eta_0^T + g^{-1}(Y_1)\eta_1^T)(\eta_0\eta_0^T + \eta_1\eta_1^T)^{-1} \end{aligned} \quad (21)$$

假设 $M_1 = \eta_0\eta_0^T + \eta_1\eta_1^T$, $M_0 = \eta_0\eta_0^T$, 则式 (21) 可变形为如下形式。

$$\begin{aligned} W_1 &= (g^{-1}(Y_0)\eta_0^T M_0^{-1} M_0 + g^{-1}(Y_1)\eta_1^T) M_1^{-1} = \\ &(W_0 M_0 + g^{-1}(Y_1)\eta_1^T) M_1^{-1} = \\ &(W_0(M_1 - \eta_1\eta_1^T) + g^{-1}(Y_1)\eta_1^T) M_1^{-1} = \\ &(W_0 M_1 + (g^{-1}(Y_1) - W_0\eta_1)\eta_1^T) M_1^{-1} = \\ &W_0 + (g^{-1}(Y_1) - W_0\eta_1)\eta_1^T M_1^{-1} \end{aligned} \quad (22)$$

根据复合矩阵 W_1 可以求出输出权值矩阵 β_1 、输入层与输出层之间的权值矩阵 ω_1^{oi} 和输出层阈值矩阵 c_1 , 如式 (23) 所示。

$$\begin{cases} \omega_1^{oi} = W_1(1:l, 1:n) \\ \beta_1 = W_1(1:l, (n+1):(n+m)) \\ c_1 = W_1(1:l, n+m+1) \end{cases} \quad (23)$$

当第 k 个样本数据到来时, 输入权值复合矩阵按式 (24) 计算。

$$\tilde{\omega}_{k+1} = \begin{cases} \tilde{\omega}_k, & \text{rank}(\Delta x_{k+1}) = 0 \\ \tilde{\omega}_k + (p\Delta x_{k+1} - \tilde{\omega}_k \Delta \tilde{x}_{k+1}) \tilde{x}_{k+1}^T K_{k+1}^{-1}, & \text{rank}(\Delta x_{k+1}) = 1 \end{cases} \quad (24)$$

基于 Woodbury 公式, 第 k 个样本数据对应的复合矩阵和输出权值矩阵分别表示如下。

$$W_{k+1} = W_k + (g^{-1}(Y_{k+1}) - W_k \eta_{k+1}) \eta_{k+1}^T M_{k+1}^{-1} \quad (25)$$

$$\begin{cases} \omega_{k+1}^{oi} = W_{k+1}(1:l, 1:n) \\ \beta_{k+1} = W_{k+1}(1:l, (n+1):(n+m)) \\ c_{k+1} = W_{k+1}(1:l, n+m+1) \end{cases} \quad (26)$$

至此, SIQNN 的计算过程推导完成。

3 改进型量子蜂群算法

人工蜂群 (artificial bee colony, ABC) 算法是根据蜂群觅食行为提出的一种群智能优化算法, 具有良好的寻优精度和收敛速度^[10]。为了提高 ABC

算法的性能, 本文提出了一种改进型量子蜂群算法, 即量子人工蜂群 (quantum artificial bee colony, QABC) 算法。

3.1 种群个体初始化

在量子计算中, 量子位是最小的信息单位, 又称作量子比特, 其不仅具有 $|0\rangle$ 和 $|1\rangle$ 态, 而且还可以表示成 $|0\rangle$ 和 $|1\rangle$ 的线性叠加态^[11]。量子位的状态表示如下。

$$|\phi\rangle = \cos(\theta/2)|0\rangle + e^{i\varphi} \sin(\theta/2)|1\rangle \quad (27)$$

由图 3 可知, Bloch 球面上的任意一点 P 都可由参数 φ 和 θ 确定, 每个量子位中的任意一个坐标都对应球面上的 3 个圆周。根据这种机制, 对 ABC 中的个体采用量子编码, 有助于扩展全局最优解的数量, 提高种群多样性并获得全局最优解的概率^[12]。

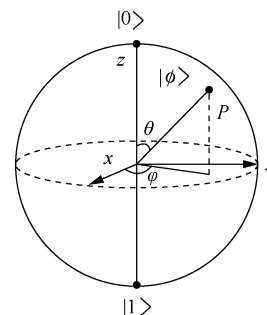


图3 量子位的 Bloch 球面表示

设 QABC 的种群规模为 m , 解空间维度为 n , 第 i 个食物源 R_i 的编码方案如式 (28) 所示。

$$R_i = \begin{bmatrix} \cos \varphi_{i1} \sin \theta_{i1} \\ \sin \varphi_{i1} \sin \theta_{i1} \\ \cos \theta_{i1} \end{bmatrix} \dots \begin{bmatrix} \cos \varphi_{ij} \sin \theta_{ij} \\ \sin \varphi_{ij} \sin \theta_{ij} \\ \cos \theta_{ij} \end{bmatrix} \dots \begin{bmatrix} \cos \varphi_{in} \sin \theta_{in} \\ \sin \varphi_{in} \sin \theta_{in} \\ \cos \theta_{in} \end{bmatrix} \quad (28)$$

根据式 (29) 确定 φ_{ij} 和 θ_{ij} 。

$$\begin{cases} \varphi_{ij} = 2\pi \times \text{rand} \\ \theta_{ij} = \pi \times \text{rand} \end{cases} \quad (29)$$

由式 (28) 可知, QABC 的种群个体分别表示如下。

$$R_{ix} = (\cos \varphi_{i1} \sin \theta_{i1}, \dots, \cos \varphi_{ij} \sin \theta_{ij}, \dots, \cos \varphi_{in} \sin \theta_{in}) \quad (30)$$

$$R_{iy} = (\sin \varphi_{i1} \sin \theta_{i1}, \dots, \sin \varphi_{ij} \sin \theta_{ij}, \dots, \sin \varphi_{in} \sin \theta_{in}) \quad (31)$$

$$R_{iz} = (\cos \theta_{i1}, \dots, \cos \theta_{ij}, \dots, \cos \theta_{in}) \quad (32)$$

3.2 解空间变换

记候选解 p_i 上的第 j 个量子位的 Bloch 坐标为

$[x_{ij}, y_{ij}, z_{ij}]^T$, 每个解空间的第 j 维的取值范围为 $[a_j, b_j]$, 则由单位空间 $I^n = [-1, 1]^n$ 映射到优化问题解空间的变换式为

$$\begin{cases} x_{ij}^{(1)} = 1/2[b_j(1 + \cos \varphi_{ij} \sin \theta_{ij}) + a_j(1 - \cos \varphi_{ij} \sin \theta_{ij})] \\ x_{ij}^{(2)} = 1/2[b_j(1 + \sin \varphi_{ij} \sin \theta_{ij}) + a_j(1 - \sin \varphi_{ij} \sin \theta_{ij})] \\ x_{ij}^{(3)} = 1/2[b_j(1 + \cos \theta_{ij}) + a_j(1 - \cos \theta_{ij})] \end{cases} \quad (33)$$

其中, $i=1, 2, \dots, n$; $j=1, 2, \dots, m$; $[a_j, b_j]$ 为第 j 个量子位的取值范围。

3.3 种群个体位置更新

在 QABC 中, 雇佣蜂和观望蜂阶段的种群个体按照式 (34) 进行修正, 该修正式可以更好地平衡探索能力和开发能力。

$$v_{ij} = x_{ij}w_{ij} + 2(a_{ij} - 0.5)(x_{ij} - x_{kj})\Phi_1 + b_{ij}(\hat{x}_j - x_{kj})\Phi_2 \quad (34)$$

式 (34) 中, a_{ij} 、 b_{ij} 为随机系数, w 为惯性权值, 可以提高解的质量和算法探索能力, 如式 (35) 所示。 Φ_1 和 Φ_2 为加速系数, 可提高算法的收敛速度, 其定义如下。

$$w_{ij} = \Phi_1 = \frac{1}{1 + \exp\left(\frac{-fit(i)}{ap}\right)^{iter}} \quad (35)$$

$$\Phi_2 = \begin{cases} 1, & \text{当前蜜蜂为雇佣蜂} \\ \frac{1}{1 + \exp\left(\frac{-fit(i)}{ap}\right)^{iter}}, & \text{当前蜜蜂为观望蜂} \end{cases} \quad (36)$$

其中, ap 为第一次迭代中最大的适应度值; $iter$ 为当前迭代次数。

4 锅炉智能燃烧优化软件设计

目前, 火力发电厂的锅炉机组控制主要是机组的协调控制, 通过调节 PID (比例-积分-微分) 控制器的参数来达到机组稳定运行的目的, 然而, 这种方法不能实现整个锅炉机组的智能优化控制。针对这种情况, 本文以 SIQNN 和改进型量子蜂群算法为核心设计了一款锅炉智能燃烧优化控制软件。

首先, 设计软件界面。本文采用 MATLAB 自带的图形用户界面 (GUI) 工具箱设计软件界面布局, 该软件可以实现数据采集、NO_x 排放浓度和煤

耗建模、NO_x 排放浓度和煤耗优化、实时显示优化前后的数据和动态显示曲线等功能。

然后, 实现数据采集功能。实现 NO_x 排放浓度和煤耗优化的前提条件是要建立 NO_x 排放浓度和煤耗的精确模型。NO_x 排放浓度和煤耗建模需要大量的历史数据, 而且尽可能包含各种工况数据, 以保证模型的泛化能力。锅炉机组的历史数据来自锅炉的 DCS, 当点击预采数据按钮时, DCS 中的锅炉机组历史数据会通过 OPC (OLE for process control) 被采集到该软件中, 用于 NO_x 排放浓度和煤耗建模及优化。需要说明的是, 锅炉的煤耗量是不能通过 DCS 直接获取的, 该软件以生产一度电消耗多少克煤作为煤耗量的粗略估计。

之后, 建立 NO_x 排放浓度模型和煤耗模型。建模采用样本增量量子神经网络, 采用该建模方法的优势是可以对建立的 NO_x 排放浓度和煤耗的模型进行实时的滚动修正, 以保证模型的准确性、实时性和泛化能力, 即使锅炉机组停机后重新启动, 也不会影响模型的准确性。点击导入数据及建模按钮后, 采集的数据会被处理, 一些坏值点会被剔除, 然后建立 NO_x 排放浓度和煤耗模型, 并保存模型参数。

最后, 优化 NO_x 排放浓度和煤耗模型, 并动态显示优化曲线。基于建立的 NO_x 排放浓度和煤耗模型, 采用改进型量子蜂群算法优化锅炉的可调节运行参量, 如给煤量、一次风压、二次风量和二次风门开度等。优化后的参数以指令的形式通过 OPC 回送给锅炉机组的 DCS, 使锅炉以优化后的数据运行, 从而达到节能减排的目的。与此同时, 在软件主界面中分别动态显示优化前后的 NO_x 排放浓度和煤耗曲线, 能够观察优化效果。除此之外, 当点击二次风门开度按钮时, 会显示优化前后的风门对比。

5 锅炉智能燃烧优化软件的应用

为了验证燃烧优化软件的有效性, 本文将该锅炉燃烧优化软件应用于某热电厂的 3 号机组 330 MW 煤粉锅炉机组上。该锅炉机组的锅炉出口排出的 NO_x 浓度过高, 导致脱硝的过程中使用了大量的氨液, 使得发电成本增加, 且存在氨逃逸现象。为了降低发电成本和降低 NO_x 排放浓度, 将锅炉燃烧优化软件外挂于 DCS 之中。由于该软件是基于 MATLAB 平台开发的, 因此在工控机上需同时安装 MATLAB 软件和锅炉燃烧优化软件。

工控机的基本配置为：CPU 主频为 3.20 GHz，内存为 8 GB，硬盘存储容量为 1 TB，操作系统为 Windows 7。

对于煤粉锅炉来说，影响 NO_x 排放浓度的主要因素有给煤量、一次风量、二次风量以及各层二次风门开度等。因此，本文主要采集影响 NO_x 排放浓度和煤耗各个因素的 46 个变量。SIQNN 的隐层节点个数设置为 40 个，隐层激励函数设置为“Sigmoid”函数，输出层节点设置为 2 个，两个输出量分别为 NO_x 排放浓度和煤耗值。QABC 的种群规模设置为 40，最大迭代次数设置为 50，“Limit”值设置为 20。软件运行效果如图 4 所示。

图 4 中，蓝色实线表示优化前的煤耗值，红色实线表示优化后的煤耗值，绿色实线表示优化前的 NO_x 排放浓度，黑色实线表示优化后的 NO_x 排放浓度。从图 4 中可以看出，通过燃烧优化软件优化后，锅炉的煤耗整体有所下降， NO_x 排放量有不同程度的降低。机组负荷为 299 MW 时，优化后的锅炉煤耗率下降了约 2.2%，优化后的 NO_x 排放浓度下降了约 56 mg/Nm^3 ，说明燃烧优化软件是有效的，达到了节能减排的目标。

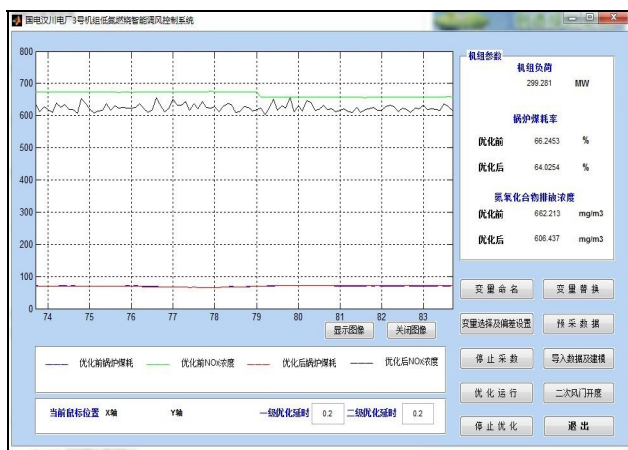


图 4 锅炉智能燃烧优化软件运行结果

为了进一步检验智能燃烧优化软件的效果，在现场又进行了机组变负荷运行试验，实验结果如图 5 所示。由图 5 可以看出，当机组负荷变化时，燃烧优化软件仍可以得到良好的优化控制效果。

6 结束语

为了实时优化燃煤锅炉的燃烧工况并达到节能减排的目的，本文提出了一种滚动优化建模方法和一种改进型量子蜂群算法。基于提出的两种方

法，设计了一款火电厂燃煤锅炉智能燃烧优化软件。通过在某 330 MW 煤粉锅炉上的应用，验证了两种方法的有效性。实验结果显示：该软件可以满足机组大范围变负荷深度调峰的要求，不同负荷下的 NO_x 排放浓度和煤耗都有较大的下降。

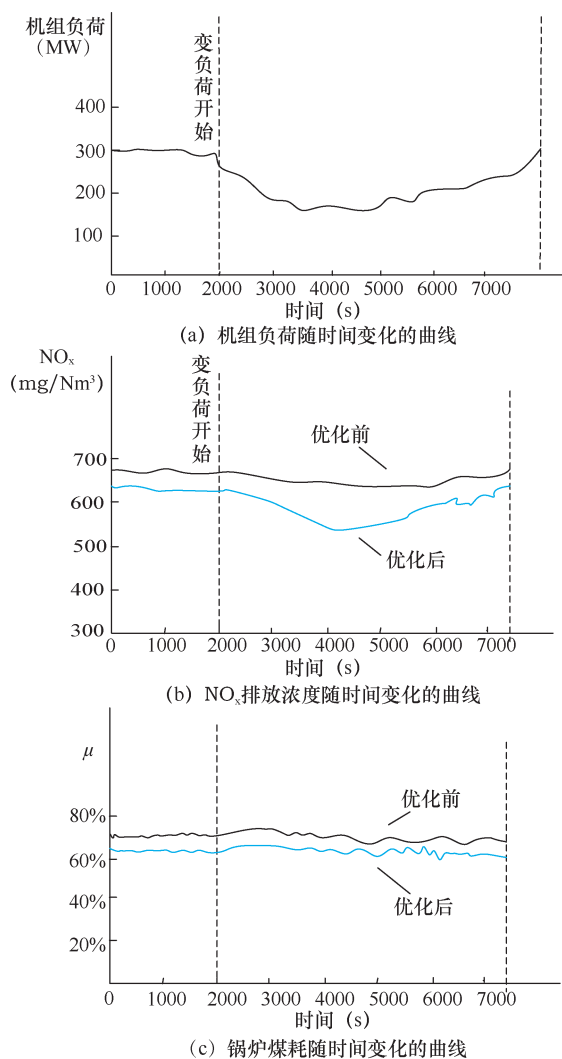


图 5 机组变负荷运行试验结果

本文所提的滚动优化建模算法还可以用于解决 UCI 数据集的分类和回归问题；改进型量子蜂群算法还可用于解决基准测试函数问题。

参考文献：

- [1] 马云鹏. 极端学习机和教与学优化算法改进及其在锅炉燃烧优化中的应用研究[D]. 秦皇岛: 燕山大学, 2018.
MA Y P. Improvement of extreme learning machine and teaching learning based optimization algorithms and its application in boiler combustion optimization[D]. Qinhuangdao: Yanshan University, 2018.
- [2] 顾燕萍, 赵文杰, 吴占松. 基于最小二乘支持向量机的电站锅炉燃烧优化[J]. 中国电机工程学报, 2010, 30(17): 91-97.

- GU Y P, ZHAO W J, WU Z S. Combustion optimization for utility boiler based on least square-support vector machine[J]. Proceedings of the CSEE, 2010, 30(17): 91-97.
- [3] 王春林, 周昊, 李国能, 等. 大型电厂锅炉 NO_x 排放特性的支持向量机模型[J]. 浙江大学学报:工学版, 2006, 40(10):1787-1791.
- WANG C L, ZHOU H, LI G N, et al. Support vector machine modeling on NO_x emission property of high capacity power station boiler[J]. Journal of Zhejiang University (Engineering Science), 2006, 40(10): 787-1791.
- [4] 许昌, 吕剑虹, 郑源, 等. 以效率 and 低 NO_x 排放为目标的锅炉燃烧整体优化[J]. 中国电机工程学报, 2006, 26(4): 46-50.
- XU C, LV J H, ZHENG Y, et al. A boiler combustion global optimization on efficiency and low NO_x emissions object[J]. Proceedings of the CSEE, 2006, 26(4): 46-50.
- [5] 赵敏华, 胡毅, 李金, 等. 使用博弈差分算法的电站锅炉高效低污染燃烧均衡优化[J]. 化工学报, 2017, 68(6): 2455-2464.
- ZHAO M H, HU Y, LI J, et al. Equilibrium optimization for high efficiency and low pollution combustion of power-generation boilers using game differential evolution algorithm[J]. SIESC Journal, 2017, 68(6): 2455-2464.
- [6] 张文广, 孙亚洲, 刘吉臻, 等. 基于自适应模糊推理辨识方法和果蝇优化算法的 CFB 锅炉燃烧优化[J]. 动力工程学报, 2016, 36(2): 84-90.
- ZHANG W G, SUN Y Z, LIU J Z, et al. Combustion optimization of OFB boilers based on adaptive fuzzy inference method and fruit fly optimization algorithm[J]. Journal of Chinese Society of Power Engineering, 2016, 36(2): 84-90.
- [7] 王东风, 刘千, 韩璞, 等. 基于大数据驱动案例匹配的电站锅炉燃烧优化[J]. 仪器仪表学报, 2016, 37(2): 420-428.
- WANG D F, LIU Q, HAN P, et al. Combustion optimization in power station based on big data-driven case-matching[J]. Chinese Journal of Scientific Instrument, 2016, 37(2): 420-428.
- [8] LIU C Y, CHEN C, CHANG C T, et al. Single-hidden-layer feed-forward quantum neural network based on Grover learning[J]. Neural Networks, 2013, 45(5):144-150.
- [9] 李盼池. 量子计算及其在智能优化与控制中的应用[D]. 哈尔滨: 哈尔滨工业大学, 2009.
- LI P C. Quantum computation with applications to intelligent optimization and control[D]. Harbin: Harbin Institute of Technology, 2009.
- [10] 魏红凯. 人工蜂群算法及其应用研究[D]. 北京: 北京工业大学, 2012.
- WEI H K. Artificial bee colony algorithm and its application[D]. Beijing: Beijing University of Technology, 2012.
- [11] 高宽云, 赵翠兰. 量子环中量子比特的性质[J]. 物理学报, 2008, 57(7): 4446-4449.

GAO K Y, ZHAO C L. Properties of qubits in quantum rings[J]. Acta Physica Sinica, 2008, 57(7): 4446-4449.

- [12] 易正俊, 何荣花, 侯坤. 量子位 Bloch 坐标的量子人工蜂群优化算法[J]. 计算机应用, 2012, 32(7): 1935-1938.

YI Z J, HE R H, HOU K. Quantum artificial bee colony optimization algorithm based on Bloch coordinates of quantum bit[J]. Journal of Computer Applications, 2012, 32(7): 1935-1938.

[作者简介]



牛培峰 (1958—), 男, 吉林舒兰人, 燕山大学教授, 主要研究方向为人工智能技术、复杂系统过程控制。



马云鹏 (1989—), 男, 河北沧州人, 博士, 天津商业大学讲师, 主要研究方向为机器学习、模式识别、复杂系统过程控制。



张欣欣 (1991—), 女, 河北邢台人, 燕山大学博士生, 主要研究方向为神经网络的稳定性分析、模式识别、机器学习等。



胡晓宾 (1983—), 男, 内蒙古乌兰察布人, 燕山大学博士生, 主要研究方向为人工智能、数据挖掘与建模。