

基于 DDPG 深度强化学习的电站 脱硝过程优化控制

林康威¹, 肖红¹, 姜文超¹, 杨建仁², 熊广思¹, 黄冠儒¹

(1. 广东工业大学 计算机学院, 广州 510006; 2. 广州云硕科技发展有限公司, 广州 511458)

摘要: 针对选择性催化还原 (SCR, selective catalytic reduction) 脱硝系统脱硝过程存在非线性、多工况等复杂特点, 提出一种基于 MiniBatchKMeans 聚类与 Stacking 模型融合的 SCR 脱硝过程 NO_x 预测方法; 该方法通过应用 MiniBatchKMeans 聚类算法对训练集进行工况聚类与划分优化, 建立基于 XGBoost、随机森林、LightGBM 以及线性回归的 Stacking 融合框架预测模型 (Stacking-XRLL), 实现电站 SCR 系统多变工况下 NO_x 排放的精准预测; 以广东某电站 SCR 系统脱硝过程中 NO_x 排放数据为例进行建模仿真与实验, 结果表明与单一建模方法多层前馈神经网络 (BP)、长短期记忆神经网络 (LSTM) 以及门控循环单元神经网络 (GRU) 相比, Stacking-XRLL 建模方法的平均预测精确度达到了 99%, 并最终结合建立好的深度确定性策略梯度 (DDPG) 强化学习模型, 实现电站 SCR 脱硝过程的参数优化控制。

关键词: 多工况; MiniBatchKMeans 聚类; Stacking-XRLL; DDPG 算法; 优化控制

Optimal Control of Denitrification Processes in Coal-Fired Power Plants Based on Deterministic Policy Gradients with Deep Reinforcement Learning

Lin Kangwei¹, Xiao Hong¹, Jiang Wenchao¹, Yang Jianren², Xiong Guangsi¹, Huang Guanru¹

(1. School of Computer Science and Technology, Guangdong University of Technology, Guangzhou 510006, China;

2. Guangzhou Yunshuo Technology Development Co., Ltd., Guangzhou 511458, China)

Abstract: A method for NO_x prediction in SCR denitration based on the fusion of MiniBatchKMeans clustering and stacking model is proposed to address the complex characteristics of the denitration process of selective catalytic reduction (SCR) denitration system, such as non-linearity and multiple working conditions. The method applies the MiniBatchKMeans clustering algorithm to the training set for work condition clustering and partitioning optimization, and establishes the stacking fusion framework prediction model (Stacking-XRLL) based on XGBoost, Random Forest, LightGBM and linear regression to achieve accurate NO_x emission prediction under multi-variable work conditions in power station SCR systems. The modeling simulations and experiments were carried out with NO_x emission data from the denitrification process of a power station SCR system in China. The results show that the Stacking-XRLL modeling method achieves an average prediction accuracy of 99% compared to the single modeling methods of the multilayer back propagation neural network (BP), long-short term memory neural network (LSTM) and gate recurrent unit neural network (GRU). The final combination of the established deep deterministic policy gradient (DDPG) reinforcement learning model enables the optimal control of the SCR denitrification process in a power station.

Keywords: multi-work condition; minibatchkmeans clustering; Stacking-XRLL; DDPG algorithm; optimal control

0 引言

近年来, 随着新能源产业迅速发展, 我国能源结构不断发生调整, 煤在能源消耗中所占的份额有所减少, 但预计在未来很长的一段时间内, 燃煤发电产业仍占据着主导地位^[1]。燃煤发电会产生污染有害气体, 其中含有 SO₂、NO、NO₂ 等。因此, 目前在火力发电厂实现控制 NO_x 排放含量, 常采用的成熟技术手段有 SCR (选择性催化还原

法) 和采用传统的 PID 控制来实现喷氨以达到脱硝优化的目的^[2-4]。而实现精准建立 SCR 脱硝过程参数与 SCR 脱硝出口 NO_x 排放浓度之间的映射关系是优化脱硝控制系统的关键基础。随着近年来人工智能算法技术的成熟, 在现有的电站 SCR 脱硝系统出口 NO_x 排放浓度的预测研究中, 大多学者分别从机理建模与数据驱动的方法进行探究。其中, 姚楚等^[5]通过 SCR 脱硝系统的化学反应机理建立 SCR 动态预测模型, 最终实验结果表明机理建模实现对脱硝系统的

收稿日期: 2021-12-07; 修回日期: 2022-01-02。

基金项目: 国家自然科学基金—广东省联合基金项目(U2001201); 广东省基础与应用基础研究基金项目(2020B1515120010)。

作者简介: 林康威(1997-), 男, 广东湛江人, 硕士, 主要从事工业大数据与数据挖掘方向的研究。

肖红(1972-), 女, 湖北咸宁人, 副教授, 硕士生导师, 主要从事智能制造与工业大数据、机器学习与人工智能等方向的研究。

通讯作者: 姜文超(1977-), 男, 山东潍坊人, 副教授, 硕士生导师, 主要从事大数据、智能控制系统与复杂网络等方向的研究。

引用格式: 林康威, 肖红, 姜文超, 等. 基于 DDPG 深度强化学习的电站脱硝过程优化控制[J]. 计算机测量与控制, 2022, 30(10): 132-139.

喷氨量控制效果优于传统的 PID 控制器的方法。但是, 通过机理建模的方式, 需要以研究对象为核心, 根据化学反应建立数学守恒关系式, 而燃煤电厂脱硝过程是一个复杂、非线性和多变量耦合的系统, 导致机理建模很难精准描述。而相对于机理建模的方法, 通过数据驱动建模的方式, 不需要深入研究对象机理反应过程, 只需以数据为驱动, 通过建立人工智能算法构建预测模型。铉佳欢等^[6]利用 BP 神经网络模型应用在 SCR 脱硝系统中, 实现预测 SCR 脱硝出口 NO_x 浓度, 从而使喷氨量得到精准控制, 与传统 PID 控制器方式相比, BP 神经网络能够很好对脱硝系统进行有效地控制, 但是其模型的泛化性有待提高。温鑫等^[7]通过构建深度双向 LSTM 神经网络模型, 实现电站 SCR 脱硝系统的出口 NO_x 排放预测, 实验结果显示与传统的 BP 神经网络模型相比较, 误差精度下降了约 5%, 但是双向 LSTM 神经网络模型结构复杂, 且模型需要优化的超参数较多。丁续达等^[8]基于最小二乘支持向量机 LSSVM 模型, 实现 SCR 脱硝系统在线 NO_x 预测, 但是模型的预测精度和泛型性上还未能达到实际工业生产的需求。虽然上述的方法不依赖于过程的结构与机理, 适合非线性强, 过程复杂的预测对象, 但针对火力发电站 SCR 系统中普遍存在着多参数耦合、调负荷、多工况等情形, 单一模型的预测精度很难达到实际应用于工业领域生产的需求。因此, 针对电站脱硝系统在多参数、多变工况条件下 NO_x 排放预测精度较低的问题, 提出基于 MiniBatchKMeans 聚类与 Stacking 模型融合的 SCR 脱硝过程 NO_x 预测方法。首先对 SCR 脱硝系统的各运行工况进行聚类分析, 然后在聚类划分基础上, 在多工况样本集以及在同工况样本集上, 利用 Stacking-XRLL 多模型融合预测模型对电站脱硝系统出口 NO_x 浓度进行预测。实验研究结果显示, 该模型在多工况下预测精度远优于 BP、LSTM、GRU 神经网络模型, 平均精度达到 99%。

另外, 实现脱硝系统出口氮氧化物超低排放是电站优化控制的重要手段。由于燃煤电厂脱硝过程的 NO_x 排放受机组负荷、喷氨质量流量、SCR 入口烟气 O₂ 量、SCR 入口烟气温度等运行参数影响。因此, 要实现 SCR 脱硝系统的 NO_x 超低排放控制, 首先需构建 SCR 脱硝系统可控运行参数与 SCR 出口 NO_x 排放的映射关系模型^[9-11], 然后再建立含有约束条件的目标优化函数, 最后基于遗传或粒子群优化算法对目标函数进行寻优^[12-15], 在满足国家要求 NO_x 排放浓度低于 50 mg · m⁻³ 约束条件下, 以获取 SCR 脱硝过程各可控运行参数的最优值。但是, 采用传统遗传和 PSO 优化算法存在收敛性不足以及局部最优解。符基高等^[16]基于 LSTM 时间循环神经模型结合深度强化学习 A3C 算法, 实现燃煤电厂 SCR 脱硝效率的控制策略。但是 LSTM 神经网络模型与 A3C 深度强化学习算法相结合之后, 存在模型训练速度慢, 且优化得到的是局部最优解, 并且评价策略通常不是非常高效, 并且有很高的偏差。因此, 在同时兼顾考虑烟气 NO_x 超低排放与脱硝效率之间的关系, 基于 MiniBatchKMeans 聚类与 Stacking 模型融合的 SCR 脱硝过程建模方法, 并利用深度确定性策略梯度 DDPG 算法对参数寻优, 为实现

现场实时优化控制奠定重要的理论基础。

1 基于 MiniBatchKMeans 聚类与 Stacking 的多模型融合算法设计

1.1 算法理论介绍

1.1.1 MiniBatchKMeans 聚类算法

MiniBatchKMeans 算法是 K-Means 算法的变种, 采用随机产生的小批量数据子集进行聚类, 大大减少了计算时间, 因此当运用在大数据集样本上时, MiniBatchKMeans 能够保持聚类准确性并可以大幅度降低计算时间。

MiniBatchKMeans 算法流程伪代码如下:

```
function MiniBatchKMeans(输入数据, 中心点个数 K) {
    获取输入数据的维度 D 和个数 N;
    随机生成 K 个 D 维的初始质心;
    while(算法未收敛) {
        从原始集随机抽取 N 个样本构建小批量样本集;
        对 N 个点: 计算每个点属于哪一类;
        对于 K 个数据中心点:
            (1) 找出所有属于自己这一类的所有数据点;
            (2) 将自己的坐标值修改为这些数据点的中心点坐标;
        }
        输出结果;
    }
}
```

两个样本点 $a = (a_1, a_2, a_3, \dots, a_n)$ 和 $b = (b_1, b_2, b_3, \dots, b_n)$ 之间距离计算如式 (1) 所示:

$$\text{dist} = \sqrt{\sum_{i=1}^n (a_i - b_i)^2} \quad (1)$$

第 i 个类中心计算公式如式 (2):

$$c_{i_q} = \frac{1}{N_i} \cdot \sum_{x \in c_i} x \quad (2)$$

其中: c_{i_q} 表示第 i 个类的类中心, N_i 表示第 i 个类中的元素个数, C_i 表示第 i 个类。

加入批量大小为 batch 的小批量样本集 $X = \{X_1, X_2, X_3, \dots, X_{\text{batch}}\}$ 后的类中心为 c_{i_q} , 计算方式如式 (3):

$$c_{i_q} = \frac{1}{N_i + \text{batch}} \cdot \sum_{x \in c_i \cup X} x \quad (3)$$

另外, 使用误差的平方和作为度量聚类质量的目标函数 func, 定义如式 (4):

$$\text{func} = \sum_{i=1}^k \cdot \sum_{x \in C_i} \text{dist}(c_{i_q}, x)^2 \quad (4)$$

1.1.2 XGBoost 算法

极端梯度提升 (XGBoost, extreme gradient boosting) 是 Tianqi Chen 在 2016 年提出的基于 Boosting Tree 模型的分布式学习框架, 该模型的基础学习器为决策树。与传统的 Boosting 树模型不同的是, 传统树模型只使用一阶导数信息, 当训练 n 棵树时, 由于使用前 $n-1$ 棵树的残差, 因此很难实现分布式训练, 而 XGBoost 对损失函数进行了二阶泰勒展开, 它可以自动使用 CPU 的多线程进行分布式计算。另外, 在目标函数中引入正则项, 以避免模型过拟合, 提高泛化性。假设有一个数据集 $D, D = \{(x_i, y_i) : i = 1 \dots n, x_i \in R^m, y_i \in R\}$, 则可以得到 n 个观测值, 每个观测值有 m 个特征以及相应的变量 y 。因此, 广义模型定义如下:

$$\hat{y}_i = \varphi(x_i) = \sum_{k=1}^K f_k(x_i) \quad (5)$$

在式 (5) 中, f_k 表示的是一个回归树, $f_k(x_i)$ 表示第 k 棵树对数据中的第 i 个观察值给出的分数。为实现目标函数 f_k , 应最小化以下正则项目标函数。

$$L(\varphi) = \sum_i l(y_i, \hat{y}_i) + \sum_k \Omega(f_k) \quad (6)$$

其中: l 是损失函数, 为防止模型过拟合, 惩罚项中 Ω 应包括以下项:

$$\Omega(f_k) = \gamma T + \frac{1}{2} \lambda \|w\|^2 \quad (7)$$

其中: γ 和 λ 分别表示叶子数量 T 和叶子权重 w 的惩罚参数。 $\Omega(f_k)$ 目的是为了防止模型过拟合而简化该算法生成的模型。

为使目标函数最小化, 采用迭代法。在第 j 次迭代中添加 f_k , 以最小化以下目标函数:

$$L^j = \sum_{i=1}^n l(y_i, \hat{y}_i^{j-1} + f_j(x_j)) + \Omega(f_j) \quad (8)$$

使用泰勒展开式来简化上述函数, 并推导出从给定节点分割树后的损失函数:

$$L_{split} = \frac{1}{2} \left[\frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} + \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma \quad (9)$$

其中: I 是当前节点中可用观测值的子集, I_L, I_R 是分割后左右节点中可用观测值的子集。函数 g_i 和 h_i 的定义如下:

$$g_i = \partial_{y_i} (j-1) \ell(y_i, \hat{y}_i^{(j-1)}) \quad (10)$$

$$h_i = \partial_{y_i}^2 (j-1) \ell(y_i, \hat{y}_i^{(j-1)}) \quad (11)$$

1.1.3 Light GBM 算法

Light GBM 算法是基于 GBDT (gradient boosting decision tree, 梯度提升决策树) 模型提出的^[17]。虽然 GBDT 在很多机器学习任务上都取得了较好的学习效果, 但近年来随着数据量的增长, 传统的 GBDT 算法在构建决策树时需要找到最优的分割点, 一般的方法是对特征值进行排序, 然后枚举所有可能的特征点。但是此种方法不仅在时间性能上表现较差, 而且需要很大的内存。因此, GBDT 算法面临着精度和效率性能的问题急需解决。

Light GBM 算法使用了改进的直方图算法, 它将连续的特征值划分为 k 个区间, 在 k 个值中选择划分点。因此, Light GBM 算法在训练速率和内存占用率上都优于传统的 GBDT 树模型。同时, 决策树是一个弱分类器, 使用直方图算法会有正则化效果, 可以有效防止过拟合。在减少更多误差方面, Light GBM 算法采用 leaf-wise 生成策略。另外, 在减少特征数量方面, 传统采用的方法是 PCA, PCA 一般用于特征冗余的情况下, 因此有一定的局限性。Light GBM 算法使用的 EFB 算法将高维数据的特征放在一个稀疏的特征空间中, 以避免计算冗余特征, 并根据算法构造直方图, 可以加快计算的速度。综合所述, Light GBM 算法在不降低预测准确率的同时, 加快预测速度, 并降低内存占用。

1.1.4 线性回归算法

线性回归分析是机器学习中的一种统计方法, 可分为简单线性回归和多元线性回归, 用于估计一个或多个输入变量和输出变量之间的关系。线性回归用直线模拟输入变量 x 和输

出变量 y 之间的关系。

一次方程定义如式 (12):

$$y = \beta_0 + \beta_1 x \quad (12)$$

其中: 参数 β_0 和 β_1 是回归系数。而模型的拟合度衡量标准, 即它对输出变量 y 的在 n 个数据点上 ϵ_i 的误差大小。

$$\epsilon_i = y_i - \hat{y}_i \quad (13)$$

式中, \hat{y}_i 为预测值, y_i 为特征值, ϵ_i 是数据点 i 的预测值与真实值之间差值。

为评估回归模型的回归预测精度与真实值之间的误差, 回归模型常用最小二乘法 (LSM, the least square method,) 估计进行拟合, 找到误差平方和最小时的最佳拟合曲线或直线, 即最小化。

$$S(\beta_0, \beta_1) = \sum_{i=1}^n \epsilon_i^2 = \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2 \quad (14)$$

假设 β_0 和 β_1 的最小二乘估计值为 $\hat{\beta}_0$ 和 $\hat{\beta}_1$, 必须满足以下条件。

$$\frac{\partial S}{\partial \beta_0} = -2 \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = 0 \quad (15)$$

$$\frac{\partial S}{\partial \beta_1} = -2 \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) x_i = 0 \quad (16)$$

对上述两个方程进行化简, 可以得到:

$$\hat{\beta}_0 = \frac{1}{n} \sum_{i=1}^n y_i - \hat{\beta}_1 \frac{1}{n} \sum_{i=1}^n x_i = \bar{Y} - \hat{\beta}_1 \bar{X} \quad (17)$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n y_i x_i - \frac{1}{n} (\sum_{i=1}^n y_i) (\sum_{i=1}^n x_i)}{\sum_{i=1}^n x_i^2 - \frac{1}{n} (\sum_{i=1}^n x_i)^2} = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2} \quad (18)$$

其中: \bar{X} 为 X_i 的均值, \bar{Y} 为 Y_i 的均值。

1.2 Stacking 模型融合

Stacking 是一种分层模型集成框架, 在基于 Stacking 的集成学习模式下, 通过融合多个机器学习算法的方式来提高整体模型的预测精度^[18]。因此, 在综合考虑 Stacking 模型融合算法的预测精度与训练性能, 将 Stacking 模型融合框架划分为两层: 第一层选择预测精度较高的 XGBoost、RandomForest 算法以及性能优异且时间复杂度较低的 LightGBM 算法模型作为基学习器; 第二层, 采用泛化能力较强和稳健性较好的线性回归算法作为元学习器, 如图 1 所示。

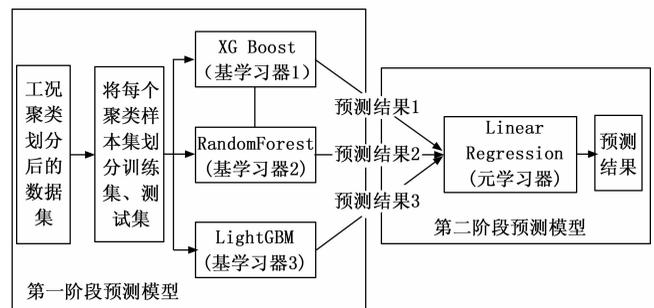


图 1 Stacking 模型融合架构图

针对电站锅炉脱硝系统在多变的工况环境条件下往往呈现出复杂的、大幅度滞后等特性, 而单一模型在一定程度上

很难准确地描述具有复杂的、非线性的火力发电厂锅炉脱硝系统 NO_x 排放问题, 导致模型预测精度不高。因此, 为了提高电站锅炉脱硝系统在多变的工况条件下 NO_x 排放预测的精度, 提出了一种基于 MiniBatchKMeans 聚类与 Stacking 多模型融合框架的电站脱硝过程建模方法, 如图 2 所示, 其建模步骤如下: 1) 将从 DCS 采集的数据集进行预处理, 其中包括剔除异常值样本和筛选稳态工况, 并按照一定比例 (4: 1) 来划分训练集与测试集; 2) 利用 MiniBatchKMeans 算法对训练集参数进行工况聚类和划分, 保存最优的轮廓系数和聚类中心, 得到 C_i 个聚类样本; 3) 对这些聚类样本, 利用如图 1 所示的融合方法, 采用 XGBoost、RandomForest、LightGBM 机器学习算法作为 Stacking 模型融合框架的第一层 (基学习器), 以线性回归作为第二层 (元学习器), 构建嵌入多个机器学习模型的 Stacking 模型融合框架预测算法, 用于处理多工况下 NO_x 的预测问题。

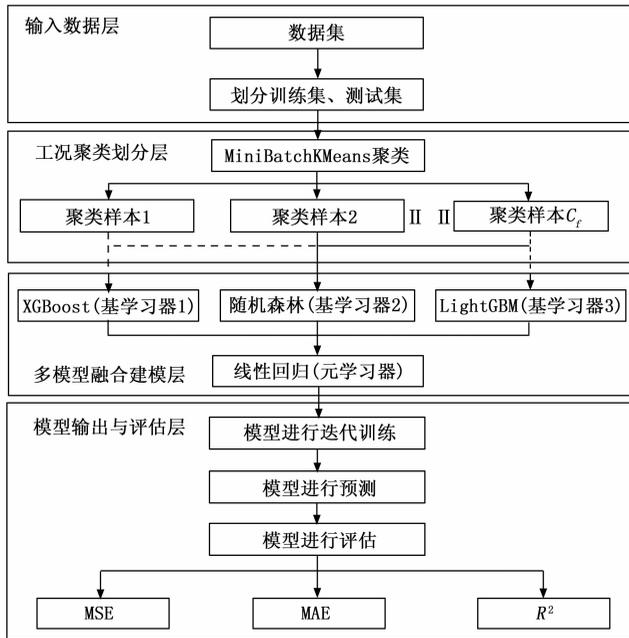


图 2 基于 MiniBatchKMeans 与 Stacking 多模型融合框架的建模流程图

2 基于深度强化学习 DDPG 算法的脱硝效率控制策略模型

2.1 DDPG 算法理论

2.1.1 基于 Actor-Critic 的深度策略梯度方法

Actor-Critic 是由 Actor 和 Critic 两个神经网络构成。Actor 负责针对 Critic 网络评价来纠正动作的偏向。Critic 负责对 Actor 生成的动作进行评分。它们整个网络的工作流程大致如下: 1) 首先 Actor 依据当前的环境生成 action; 2) 环境依据 Action 给与相应的回报 r ; 3) Critic 会对 action 进行评价; 4) Actor 会依据 Critic 的评价来调整策略, 输出新的 action; 5) Critic 会依据回报 r 来纠正评价规则。不断循环 (1) ~ (5), 直至所有的网络收敛或达到设定训练周期的阈值。

在 Actor-Critic 网络中, 通常情况下, Critic 是一个状态值函数, 在每次动作选择之后, Critic 会评估新的状态以确定

事件是否比预期的好还是坏, 这个评价就是时间差分法 (temporal difference, TD), 数学表达式如式 (19) 所示:

$$V(s_t) \leftarrow V(s_t) + \alpha[r_{t+1} + \gamma V(s_{t+1}) - V(s_t)] \quad (19)$$

其中: V 是有评判者 (Critic) 实现的值函数。TD 误差用来评估所选择动作, 即在某状态下所采取的行动。如果 TD 误差是正的, 表示未来应加强选择的倾向, 而如果 TD 是负的, 表明未来应减弱这种倾向。这种假设动作是由 Gibbs Softmax 方法产生的, 如式 (20) 所示:

$$\pi_t(s, a) = P_r\{a_t = a \mid s_t = s\} = \frac{e^{p(s,a)}}{\sum_b e^{p(s,b)}} \quad (20)$$

式中, $p(s, a)$ 是行为者 (Actor) 在时间 t 的可修改策略参数, 表示在每个状态 s 时选择每个动作 a 的倾向。对上述的加强与减弱可通过调整 $p(s, a)$ 来实现, 如式 (21) 所示:

$$p(s_t, a_t) \leftarrow p(s_t, a_t) + \beta \delta \quad (21)$$

式中, β 是一个正的步长参数, 这是一个奖赏惩罚方法。无论 TD 误差 δ 是正还是负, 都会对策略进行更改。当 δ 为正时, 增加动作的概率, δ 为负时, 减少动作的概率。

2.1.2 基于 DDPG 深度确定策略梯度方法

DDPG 算法一种强化学习框架, 基于策略梯度与 DQN 算法, DDPG 能够解决 Actor-Critic 在连续动作空间的问题。例如在 Gym 和 TORCS 领域中, DDPG 可以直接使用原始状态来学习, 并且在 Atari 领域比 DQN 使用更少的经验学习步骤^[19]。

DDPG 的核心是使用一种随机的方法来探索好的行为, 但估计一个确定性的行为策略 (如式 (22) 所示)。只需在状态空间上进行整合, 使得学习策略变得更加容易, 但它也有可能无法探索完整状态和动作空间的局限性, 为克服这个局限性, 在随机探索的加入一个噪声 N_t 。

$$a_t = \mu(s_t \mid \theta^a) \quad (22)$$

$$a_t = \mu(s_t \mid \theta^a) + N_t \quad (23)$$

DDPG 中的 Actor 和 Critic 是由神经网络设计的。Actor 网络根据确定性策略梯度规则进行更新, 而 Critic 网络则根据 TD 误差中获得梯度进行更新, 如式 (24) 所示:

$$\nabla_{\theta^a} \mu \approx E_{\mu}[\nabla_a Q(s, a \mid \theta^Q) \mid_{s=s_t, a=\mu(s_t)} \nabla_{\theta^a} \mu(s \mid \theta^a) \mid_{s=s_t}] \quad (24)$$

式中, 为得到期望值, 需要 Critic 网络在行动方面的梯度 (w, r, t) 以及 Actor 网络 (w, r, t) 和其它参数。DDPG 网络参数的更新规则, 采用小批量 (mini-batch) 数据样本, 通过最小化式 (25) 中的损失来更新 Critic 网络, Actor 网络使用采样策略梯度更新, 如式 (26) 所示:

$$L = \frac{1}{N} \sum_i (y_i - Q(s_i, a_i \mid \theta^Q))^2 \quad (25)$$

$$\text{其中: } y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1} \mid \theta^a) \mid \theta^Q)。$$

$$\nabla_{\theta^a} \mu \approx \frac{1}{N} \sum_i [\nabla_a Q(s, a \mid \theta^Q) \mid_{s=s_t, a=\mu(s_t)} \nabla_{\theta^a} \mu(s \mid \theta^a) \mid_{s=s_t}] \quad (26)$$

而目标 Actor 网络和 Critic 网络的参数更新如式 (27) 和 (28) 所示:

$$\theta^Q \leftarrow \tau \theta^Q + (1 - \tau) \theta^Q \quad (27)$$

$$\theta^a \leftarrow \tau \theta^a + (1 - \tau) \theta^a \quad (28)$$

其中: τ 是更新参数, 将其设置为 $\tau Q \ll 1$ 。

2.2 深度强化学习模型的建立

构建基于 DDPG 算法的 SCR 脱硝效率深度强化学习模型的整体框架如图 3 所示。在网络结构的设计中, Actor 网络(主网络和目标网络)和 Critic 网络(主网络和目标网络)都包含两个隐藏层网络, 每层神经元个数分别设置为 256 和 128。Actor 网络最后一层的激活函数为 tanh 函数, 使得每一层的动作输出控制在 $[-1, 1]$ 之间, 最终依据脱硝效率状态值限定范围得到脱硝系统各运行参数可控值。Critic 网络对 Actor 网络得到的脱硝系统可控参数进行评估, 采用 relu 激活函数。经过反复实验调试, DDPG 模型的学习训练周期设置为 500, Actor 网络学习率设置为 0.001, Critic 网络学习率设置为 0.002。

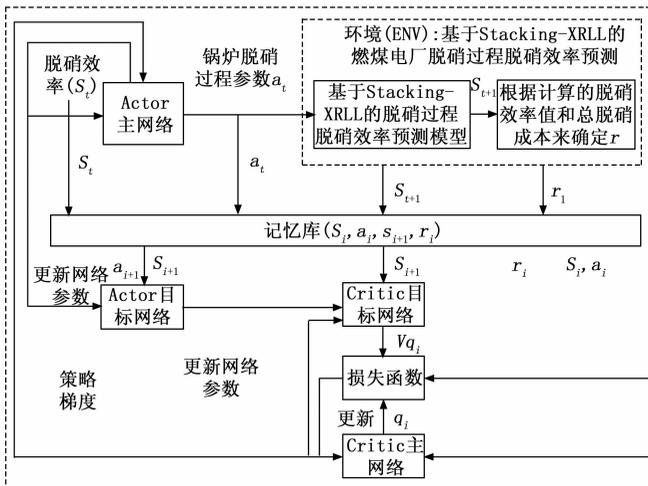


图 3 基于深度强化学习 DDPG 模型的脱硝过程优化控制架构

DDPG 算法模型选取机组负荷、喷氨质量流量、SCR 入口烟气 O_2 量、SCR 入口烟气温、SCR 入口 NO_x 质量浓度 5 个变量作为 action 动作值, 脱硝效率(计算方式如式(29)所示)作为 state 状态值, 且各参数变量取值范围设置如表 1 所示。

表 1 电站锅炉各参数运行范围

参数	符号	取值范围
机组负荷/(MW)	L	394.3474~998.8000
SCR 入口烟气温 / °C	T	302.214 1~357.200 8
SCR 入口烟气 O_2 量 / (mg · m ⁻³)	O	0.993 67~ 6.939 1
SCR 入口 NO_x 质量浓度 / (mg · m ⁻³)	$NO_{x,n}$	130.449 4~491.223 6
可控变量 喷氨质量流量 (kg / h)	NH_3	29.378 1~ 192.493 3
输出变量 SCR 出口 NO_x 质量浓度 / (mg · m ⁻³)	$NO_{x,out}$	2.159 1~95.013 6

$$\eta = \frac{I_{NOx_in} - I_{NOx_out}}{I_{NOx_in}} \times 100\% \quad (29)$$

式(29)中, η 为脱硝效率, I_{NOx_in} 为 SCR 入口 NO_x 质量浓度, I_{NOx_out} 为 SCR 出口 NO_x 质量浓度。在 SCR 脱硝系统中, 加大喷氨量, 可以提高脱硝效率, 但是过多的喷氨, 另外会造成脱硝成本的提高。而喷氨量是衡量脱硝成本的重要指标, 脱硝成本计算结果等于单位机组负荷下的喷氨量乘以相应单价。通常情况下, 一般每台锅炉配备两台脱

硝设备, 因此脱硝成本计算公式如式(30)所示:

$$T = \frac{2 \times P_{NH_3} \times M_{NH_3}}{L} \quad (30)$$

式(30)中, T 为总成本, P_{NH_3} 为喷氨量的单价(按市场价约 3 500 元/t), M_{NH_3} 为总喷氨量, L 为机组负荷。

因此, 在设置模型的奖励函数时, 应兼顾喷氨量与脱硝效率之间的平衡关系。根据专家经验, 当脱硝效率(η)处在 85%~95% 的合理范围区间, 并同时满足总脱硝成本 T 是最小化时, 应当给与奖励 (reward=10)。其余情况下, 都认为是不合理的, 应当给与惩罚 (reward=-20)。

DDPG 模型的伪代码流程如下:

随机初始化 Critic $Q(s, a | \theta^Q)$ 和 Actor $\mu(s | \theta^\mu)$ 主网络参数, 初始权重为 v^Q 和 θ^μ ;

初始化目标网络 Q' 和 μ' , 初始权重为 $\theta^Q \leftarrow \theta^Q, \theta^{\mu'} \leftarrow \theta^\mu$;

初始化记忆库缓冲区大小为 b ;

for episode = 1, ..., M do

接收一个状态值 s_t ;

for t = 1, ..., T do

基于 ϵ 贪婪算法选择一个动作值 a_t ; 以概率 ϵ 选择随机选择一个动作, 否则以 $a_t = \mu(s_t | \theta^\mu)$ 的当前策略进行选择;

执行动作 a_t , 输入到 Stacking-XRLL 模型中, 预测 SCR 出口 NO_x 浓度, 然后计算的脱硝效率 η , 最后再根据设定奖励规则, 生成回报 r_t 和新的状态值 s_{t+1} ;

将 t 时刻样本数据 (s_t, a_t, r_t, s_{t+1}) 存储到记忆库 b 中;

当记忆库的数据存满, 随机采样 N 个转换数据 (s_i, a_i, r_i, s_{i+1}) , 作为 Actor、Critic 目标网络的一个单位输入组数据集进行训练;

设置 $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1} | \theta^{\mu'})) | \theta^Q$;

通过最小化损失更新 Critic 网络参数: $L = \frac{1}{N} \sum_j (y_i - Q(s_i, a_i | \theta^Q))^2$;

使用策略梯度更新 Actor 网络参数:

$$\nabla_{\theta^\mu} \mu | s_j \approx \frac{1}{N} \sum_j \nabla_a Q(s, a | \theta^Q) |_{s=s_j, a=\mu(s_j)} \nabla_{\theta^\mu} \mu(s | \theta^\mu) | s_j$$

最后更新目标网络参数:

$$Q^Q \leftarrow \nu Q^Q + (1 - \nu) Q^Q$$

$$\theta^{\mu'} \leftarrow \nu \theta^{\mu'} + (1 - \nu) \theta^{\mu'}$$

end for

end for

根据上述的 DDPG 算法伪代码流程, 迭代训练 500 个周期, 即过程通过不断调整评判者网络参数以修正行为网络的参数, 直至 Actor 网络和 Critic 网络趋于稳定, 进而优化燃煤电厂电站锅炉脱硝过程可控运行参数, 使得基于 Stacking-XRLL 的多模型建模的 SCR 脱硝过程氮氧化物预测模型输出满足 SCR 脱硝出口 NO_x 排放浓度(低于 $50 \text{ mg} \cdot \text{m}^{-3}$)、脱硝效率处于合理范围区间内 ($85\% \leq$ 脱硝效率 $\leq 95\%$) 以及总脱硝成本 T 最小化时, 最终可以获得满足条件的最优可控动作参数集。

3 实验结果与分析

3.1 实验环境与数据

本文进行实验所需的硬件设备(计算机)配置如下: 中央处理器: Intel (R) Core (TM) i7-9750H CPU @2.60 GHz

2.59 GHz; 计算机内存: 16 GB RAM; 操作系统: Windows10-64 位; 图形处理器: NVIDIA GeForce GTX1660Ti 6 GB。

本文进行实验所需的软件平台包括: 运用 Python 编程语言; 编程环境: Python v3.7、Python IDEA; Pycharm v2020.1; Scikit-learn 库: v0.22.1; numpy: 1.19.4; pandas: 1.1.4; matplotlib: 3.3.2。

在进行实验时所需数据集是以广东某电厂 1 000 MW 电站 SCR 脱硝系统为研究对象, 根据 SCR 系统运行状况和专家经验分析, 从 DCS 信息数据采集系统中选取机组负荷、喷氨质量流量、SCR 入口烟气 O_2 量、SCR 入口烟气温度、SCR 入口 NO_x 质量浓度、SCR 出口 NO_x 质量浓度等一共 6 个特征, 数据如表 1 所示。其中可控变量: 喷氨质量流量。状态变量: 机组负荷、SCR 入口烟气 O_2 量、SCR 入口烟气温度和 SCR 入口 NO_x 质量浓度。输出变量: SCR 出口 NO_x 质量浓度。选取 2018 年 4 月 1 日—2018 年 4 月 30 日时段内 SCR 脱硝系统机组稳态运行状态数据, 每间隔为 60 s 采集一次数据, 最终取 10 000 条样本作为模型的数据集。对从 DCS 系统采集到的样本数据集进行数据预处理, 其中包括剔除异常值样本和筛选稳态工况。稳态工况可以利用滑动窗口法进行判断, 如式 (31) 所示^[20]:

$$func_{aj} = \sqrt{\frac{1}{n-1} \sum_{k=r-n+1}^r ((x_j - \bar{x}_j))^2} < \sigma_c \quad (31)$$

在式 (31) 中, 其中 $n=35$ 为窗口宽度, x_j 为归一化后的特征变量参数, 可以选择机组负荷, $\sigma_c=0.65$ 为稳态工况的阈值。

3.2 MiniBatchKMeans 工况聚类划分

从 DCS 系统采集 10 000 条稳态工况数据样本, 按照 4:1 的比例划分训练集与测试集, 同时保证训练集和测试集涵盖 SCR 系统各运行工况。经过与电厂专家交流分析后, 将从 DCS 系统采集的 6 个特征变量作为模型的输入变量, SCR 出口氮氧化物浓度作为模型的输出变量。设定初始聚类簇数值在 $[2, 11]$ 范围内, 分别计算相应值下的轮廓系数, 当聚类簇个数 $C_f=7$ 时, 总的轮廓系数最大, 此时聚类效果最好, 最终将训练集按机组负荷聚类为 7 个子簇。经过 MiniBatchKMeans 聚类所得工况聚类划分结果如表 2 所示。

表 2 工况聚类划分结果

参数	子集						
	工况 1	工况 2	工况 3	工况 4	工况 5	工况 6	工况 7
聚类中心机组负荷/MW	950.84	679.37	508.55	860.15	598.88	759.81	411.58
样本个数	4 579	1 078	795	1 022	734	1 606	186

3.3 多模型融合建模预测

对 7 个子集分别利用基于 Stacking-XRLL 多模型融合算法进行建模, 将获得的 10 000 条样本数据, 8 000 条作为训练集, 2 000 条作为测试集。最后利用所建立的模型在测试集上进行预测, 得到 SCR 脱硝出口 NO_x 排放浓度预测结果如图 4 所示。采用模型评估指标: 平均绝对误差 (MAE)、均方误差 (MSE) 和决定系数 R^2 对模型进行评价如表 3 所示。

表 3 不同工况模型预测结果性能对比

工况	MSE	MAE	R^2
未分工况	16.890 3	1.740 4	0.925 4
工况 1	0.398 6	0.044 5	0.997 4
工况 2	0.034 1	0.032 4	0.999 6
工况 3	0.007 1	0.006 7	0.999 9
工况 4	0.001 0	0.004 4	0.999 9
工况 5	0.006 3	0.007 6	0.999 9
工况 6	0.030 6	0.066 1	0.999 5
工况 7	4.016 9	1.192 0	0.999 8

由图 4 与表 3 可知, 对 SCR 系统的运行工况进行聚类划分之后, 在每一个子集工况下分别利用基于 Stacking-XRLL 模型进行预测, 实验结果表明, 未进行工况划分之前, 模型预测精度 MSE (均方误差) = 16.890 3、MAE (平均绝对误差) = 1.740 4 和 R^2 (决定系数) = 0.925 4。而锅炉运行工况进行聚类划分之后, 在各个工况下进行预测, 每一类工况下预测的精度都得到了提升, 其中每个工况下总的均方误差 MSE = 0.642 0、平均绝对误差 MAE = 0.193 3 和 $R^2 = 0.999 4$ 。

为了充分验证本文所提出的基于 Stacking-XRLL 多模型融合算法的有效性, 从 7 个工况中随机选取工况 2 下的数据集, 将其分别与单模型最优模型参数条件下的 BP 神经网络、LSTM 神经网络模型、GRU 神经网络模型进行对比实验, 如图 5 所示。其中, BP 神经网络为 3 层网络架构, 第一层有 256 个神经元, relu 为激活函数, dropout 率为 0.2; 第二层有 128 个神经元, relu 为激活函数, dropout 率为 0.3; 第三层为全连接层。LSTM 循环神经网络总共建立四层 LSTM 层, 神经元个数分别为 128、128、64 和 32, dropout 率为 0.3, tanh 为激活函数, 最后一层为全连层。GRU 神经网络总共建立 5 层 GRU 层, 神经元个数分别为 128、64、256、256 和 128, dropout 率为 0.3, tanh 为激活函数, 最后一层为全连接层。

表 4 同工况下不同算法之间的预测结果性能对比

模型	R^2	MSE	MAE
Stacking-XRLL	0.999	0.110	0.030
BP	0.983	2.732	1.161
GRU	0.896	16.438	1.357
LSTM	0.850	24.701	1.683

由图 5 与表 4 可知: 在同一工况条件下, 单模型 BP 神经网络要优于单模型 GRU 神经网络, 而单模型 GRU 神经网络要优于单模型 LSTM 神经网络, 但是基于 Stacking-XRLL 多模型融合算法, 无论是精度上还是泛化性能上都优于 BP 神经网络、GRU 神经网络、LSTM 神经网络, 其中 MSE = 0.110、MAE = 0.030 和 $R^2 = 0.999$ 。因此, 实验结果表明: Stacking-XRLL 多模型融合算法, 能够有效且精准地预测电站 SCR 系统脱硝出口 NO_x 浓度。

3.4 DDPG 优化控制的结果

由 4.3 小节的实验结果, 得出 Stacking-XRLL 模型预测

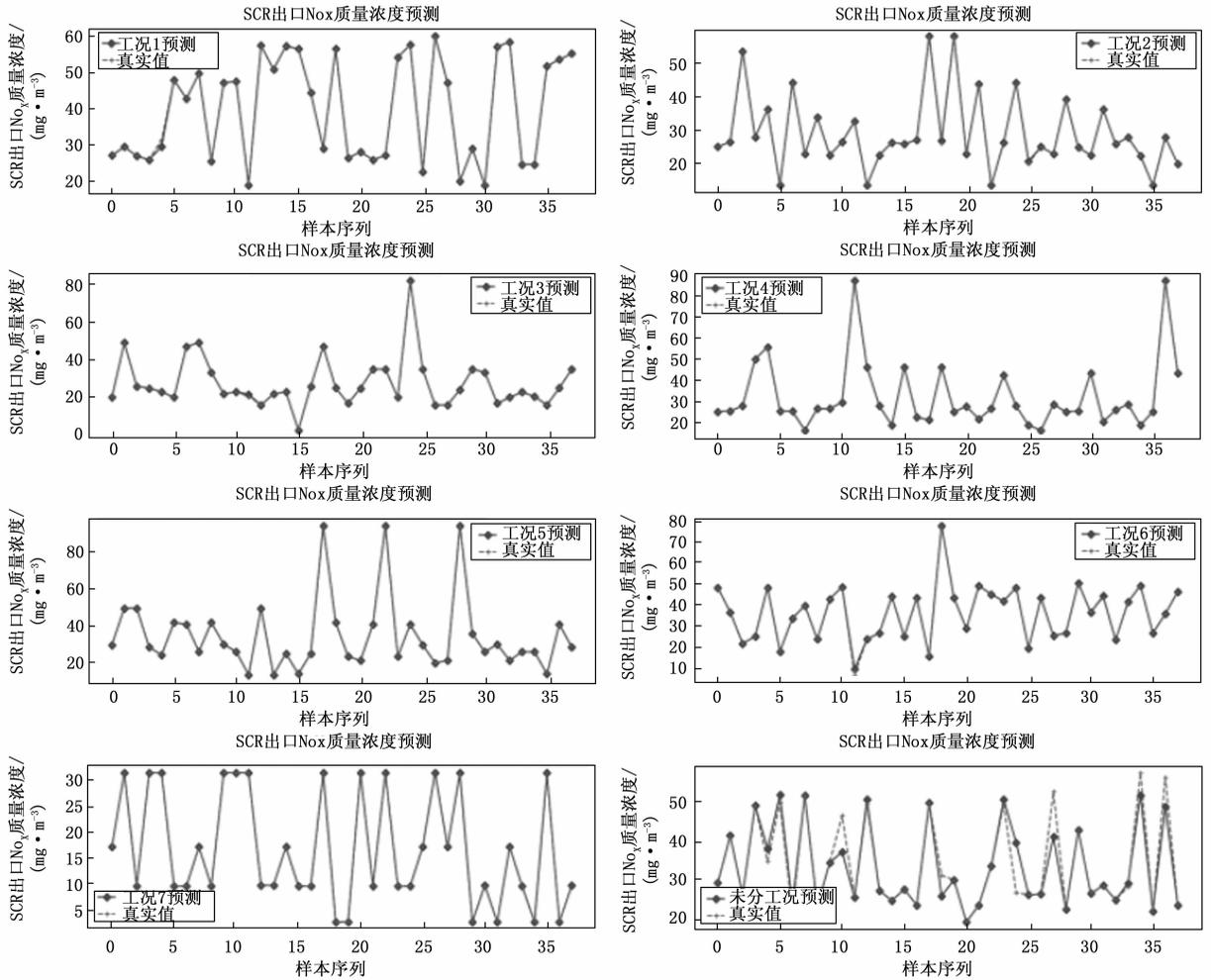


图 4 不同工况下模型的预测结果

的精度最优。因此，将 Stacking-XRLL 预测模型作为深度强化学习 DDPG 模型中的环境 (ENV, Environment)，以工况 1 作为实验的数据集，经过反复实验调试，最终确定强化学习周期设置在 500，每个周期 100 回合时，实验的收敛效果

最明显，每回合取一个预测结果。当模型迭代训练稳定时，得到实验结果如图 6 中 (a) ~ (d) 所示。

从图 6 (a) ~ (d) 实验结果可以看出，DDPG 深度学习优化控制模型的总奖励值在 200 回合后趋于稳定。即当 DDPG 深度学习模型稳定时，模型的总奖励值由一开始惩罚到奖励，不断迭代训练，最终趋于最优值稳定。此时，脱硝效率值稳定在 86% 左右，处在合理范围区间之内，且可控参数喷氨质量流量稳定在 35.657 kg/h，且经过优化之后，脱硝成本总价格降低了 27.56%。

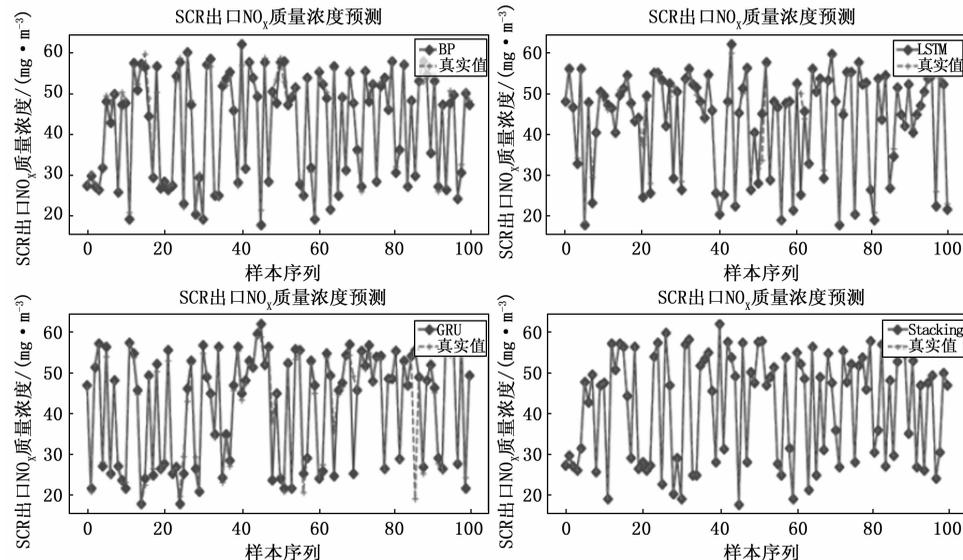


图 5 同工况下不同算法之间预测结果对比

4 结束语

脱硝效率作为衡量 SCR 脱硝系统主要指标，对脱硝系统乃至整个发电机组都有着重大影响。实现准确预测脱硝效率，能够对机组的稳定运行和优化控制起到推动作用。将机组负荷、SCR 入口烟气温度、SCR 入口烟气 O₂ 量、SCR 入

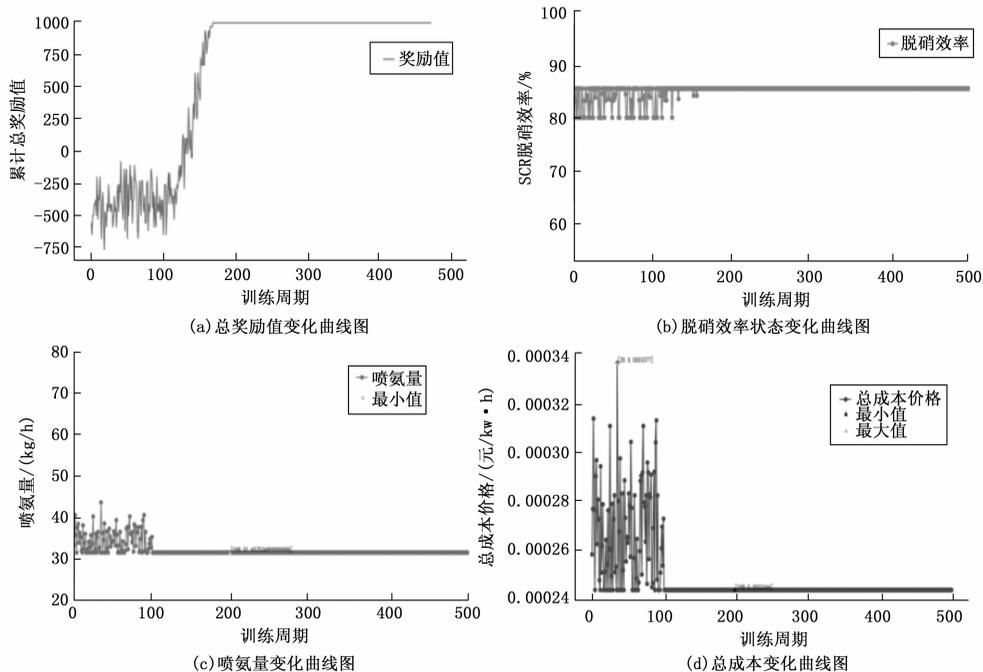


图 6 硝过程参数优化控制结果图

口 NO_x 质量浓度和喷氨质量流量等参数作为输入, 基于 Stacking-XRLL 模型融合算法, 构建深度确定性策略梯度网络优化控制模型, 实现对可调运行参数的优化, 得到不同工况下的最优操作参数值。基于某 1 000 MW 燃煤电厂机组实际运行数据进行仿真, 结果表明通过优化后机组的脱硝效率稳定在 86% 左右, 同时能满足脱硝出口 NO_x 排放浓度要求以及总脱硝成本相比未优化之前降低了 27.56%。

参考文献:

- [1] 郦建国, 朱法华, 孙雪丽. 中国火电大气污染防治现状及挑战 [C] // 中国环境保护产业协会. 第十八届中国电除尘学术会议论文集. 中国环境保护产业协会: 中国环境保护产业协会电除尘委员会, 2019: 10.
- [2] 汪立忠. 选择性催化还原法 (SCR) 催化剂在玻璃熔窑烟气脱硝中的应用研究 [J]. 玻璃与搪瓷, 2019, 47 (1): 44-48.
- [3] 张贺, 秘雪岳, 孙妍, 等. NH_3 -SCR 脱硝技术的研究现状与展望 [J]. 实验室科学, 2019, 22 (6): 6-10.
- [4] 刘晓哲. 模糊 PID 控制在 SCR 烟气脱硝系统中的应用 [J]. 电力系统装备, 2020 (7): 107-108.
- [5] 姚楚, 龙东腾, 吕游, 等. 火电厂 SCR 脱硝系统机理建模与控制研究 [J]. 热能动力工程, 2018, 33 (5): 78-84.
- [6] 铨佳欢, 邸帅. BP 神经网络算法在 SCR 脱硝系统中的应用 [J]. 电站系统工程, 2020, 36 (5): 21-24, 28.
- [7] 温鑫, 钱玉良, 彭道刚, 等. 基于深度双向 LSTM 的 SCR 系统 NO_x 排放预测模型研究 [J]. 热能动力工程, 2020, 35 (10): 57-64.
- [8] 丁续达, 金秀章, 张扬. 基于最小二乘支持向量机的改进型在线 NO_x 预测模型 [J]. 热力发电, 2019, 48 (1): 61-67.
- [9] 王博, 赵亮, 赵长春, 等. 基于 EMD-SVR 的火电厂选择

性催化还原脱硝系统出口 NO_x 浓度预测研究 [J]. 计算机测量与控制, 2020, 28 (5): 71-75, 106.

- [10] 钟用禄, 李海山, 刘发圣, 等. 基于 PCA-SVR 的燃煤锅炉 NO_x 排放预测 [J]. 热力发电, 2015, 44 (1): 87-90.
- [11] 章文涛, 张东平. 基于 IGWO-BP 的 SCR 脱硝效率软测量模型 [J]. 计算机测量与控制, 2021, 29 (10): 66-70, 76.
- [12] 朱清智, 董泽. 基于优化 PSO 算法的 SCR 脱硝控制器研究 [J]. 电力科学与工程, 2019, 35 (2): 20-26.
- [13] 杨婷婷, 白杨, 吕游, 等. SCR 脱硝系统多目标优化控制研究 [J]. 中国电机工程学报, 2021, 41 (14): 4905-4911.
- [14] 陈波, 曹歌瀚, 黄亚继, 等. 基于机器学习的燃煤锅炉燃烧效率在线计算 [J]. 洁净煤技术, 2021, 27 (4): 174-179.
- [15] 李伟, 徐强, 孔德安, 等. 电站锅炉 SCR 脱硝系统联合运行优化模型 [J]. 热力发电, 2019, 48 (6): 46-52.
- [16] FU J, XIAO H, WANG H, et al. Control strategy for denitrification efficiency of coal-fired power plant based on deep reinforcement learning [J]. IEEE Access, 2020, 8: 65127-65136.
- [17] KE G, MENG Q, FINLEY T, et al. Lightgbm: A highly efficient gradient boosting decision tree [J]. Advances in neural information processing systems, 2017, 30: 3146-3154.
- [18] 史佳琪, 张建华. 基于多模型融合 Stacking 集成学习方式的负荷预测方法 [J]. 中国电机工程学报, 2019, 39 (14): 4032-4042.
- [19] DUAN Y, CHEN X, HOUTHOOFT R, et al. Benchmarking deep reinforcement learning for continuous control [C] // International Conference on Machine Learning, PMLR, 2016: 1329-1338.
- [20] YOU L, LIU J, YANG T, et al. A novel least squares support vector machine ensemble model for NO_x emission prediction of a coal-fired boiler [J]. Energy, 2013, 55 (15): 319-329.