

# 基于自适应学习率卷积神经网络的新型配电网源网荷储无功协调优化技术

钱进宝<sup>1</sup>, 刘晓光<sup>2</sup>, 蔡玺<sup>2</sup>, 刘熠<sup>2</sup>, 戴剑丰<sup>3</sup>

(1.国网甘肃省电力公司兰州供电公司, 甘肃 兰州 730000; 2.甘肃同兴智能科技发展有限责任公司, 甘肃 兰州 730050; 3.南京邮电大学 自动化学院 人工智能学院, 江苏 南京 210023)

**摘要:** 随着“双碳”目标的推进,分布式新能源接入电网的容量大幅度提升,配电网源网荷储协调优化策略是实现分布式新能源消纳的重要方法,其中无功优化能够保证电网安全稳定运行。文章提出了一种基于自适应学习率卷积神经网络的配电网源网荷储无功协调优化技术。首先以最小网络损耗和最低电压偏移为目标,构建无功优化模型;其次利用卷积神经网络强大的非线性拟合能力,挖掘出电网运行场景和无功调压设备、储能充放电策略之间的映射关系,引入自适应学习率的方式更新网络参数,提高了网络训练效率;再次通过控制无功调压设备和储能装置充放电情况协调分布式电源出力,实现电力系统无功电压主动优化控制;最后通过IEEE33节点电网模型进行了仿真验证。结果表明,文章所提的配电网源网荷储无功协调优化方法提高了电力系统电压调节能力,为配电网安全可靠运行奠定了良好基础。

**关键词:** 分布式新能源; 源网荷储协调优化; 无功优化; 自适应学习率; 卷积神经网络

**中图分类号:** TK01; TM73 **文献标志码:** A **文章编号:** 1671-5292(2024)02-0267-09

## 0 引言

在配电网源网荷储协调优化中,无功协调优化能够保证电网安全稳定运行。传统的无功协调优化方法主要是规划方法,包含线性规划<sup>[1,2]</sup>、非线性规划<sup>[3,4]</sup>以及混合整数规划<sup>[5,6]</sup>,这些方法计算量大,过于依赖潮流模型,适应性较差。智能算法具有很强的计算能力,在电网优化配置中得到了广泛应用。常见的智能算法包括粒子群算法<sup>[7,8]</sup>、遗传算法<sup>[9,10]</sup>、模拟退火算法<sup>[11,12]</sup>、蚁群算法<sup>[13,14]</sup>等,但是这些算法容易出现计算不稳定、陷入局部最优的问题。

随着人工智能技术的快速发展,基于数据驱动的无功优化技术得到了广泛应用。文献[15]提出了基于数据驱动的风电场无功优化技术,提升了算法的优化能力。文献[16]利用负荷分布和熵权法进行无功优化,从历史数据中找出相似的负荷分布,再使用熵权法进行最优控制。文献[17]提出了基于深度置信网络的无功优化方法,建立了系统特征和优化方法之间的映射模型。文献[18]基于随机矩阵的无功协调优化方法,将提取的关

键信息与场景进行匹配,得出分配策略。然而,面对复杂的源网荷储配电网,如何准确快速地协调无功调压装置、储能设备之间的动态输出,适应电网运行状态变化和保障电网安全稳定运行,目前考虑的较少。

深度学习是人工智能技术的高阶算法,能够处理复杂的数据特征,得出输入和输出之间的隐藏映射关系<sup>[19,20]</sup>。卷积神经网络是一种典型的深度学习算法,具有很强的非线性拟合能力,其权值共享的特性能够减少参数量,具备收敛速度快、特征学习能力强等优点<sup>[21]</sup>。

本文提出了一种基于自适应学习率卷积神经网络的配电网源网荷储无功协调优化技术。以最小网络损耗和最低电压偏移为目标,构建无功优化模型,基于卷积神经网络挖掘出电网运行场景和无功优化策略之间的映射关系,引入自适应学习率提高网络训练效率。仿真验证了本文所提方法能够提高电力系统的电压调节能力。

## 1 配电网源网荷储无功优化模型

随着分布式电源比例的提升,新型源网荷

收稿日期: 2023-10-17。

基金项目: 国家自然科学基金项目(62173188)。

作者简介: 钱进宝(1985-),男,硕士,高级工程师,研究方向为配电网安全稳定运行。E-mail: qianjinbao1068@126.com

通信作者: 戴剑丰(1989-),男,博士,讲师,研究方向为新型电力系统的安全稳定分析与控制。E-mail: daijianfeng@njupt.edu.cn

储配电网无功电压问题更加突出。本文搭建了以最小网络损耗和最低电压偏移为目标的配电网源网荷储无功优化模型,通过调节储能装置充放电策略、无功调压设备出力,实现系统最优无功潮流分布。

### 1.1 分布式电源接入对配电网电压的影响

分布式电源接入配电网等值电路如图 1 所示。

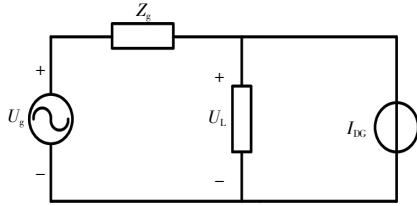


图 1 分布式电源接入等值电路图

Fig.1 Contour map of distributed power supply access

分布式电源接入前、后的负荷电压表达式分别为

$$U_{L1} = \frac{U_g Z_L}{Z_L + Z_g} \quad (1)$$

$$U_{L2} = \frac{Z_L}{Z_L + Z_g} (U_g + Z_g I_{DG}) \quad (2)$$

式中: $U_{L1}, U_{L2}$  分别为分布式电源接入前、后的负荷电压; $U_g$  为电网电压; $Z_L, Z_g$  分别为负荷、电网的阻抗; $I_{DG}$  为流过分布式电源的电流。

通过式(1),(2)可知, $U_{L2} > U_{L1}$ ,分布式电源并网电压增加。此外,分布式电源输出和负荷需求容量具有很强的不确定性,使得配电网节点电压明显增大,进而导致节点电压越限。如果分布式电源输出较大以及轻载运行,则分布式电源并网电压容易越上限;如果分布式电源输出较小以及重载运行,则分布式电源并网电压容易越下限。

### 1.2 目标函数

以整个系统网损最小、电压偏移最低为目标,将节点电压约束以惩罚函数的方式构建无功优化目标函数,某一时刻模型如下所示:

$$f(t) = \sum_{i,j \in N_l} g_{ij} (U_i^2 + U_j^2 - 2U_i U_j \cos \theta_{ij}) + \sum_{i=1}^{N_{ro}} \left( \frac{U_i - U_{iN}}{U_{iN}} \right) + \lambda \sum_{i=1}^{N_{ro}} \left( \frac{U_i - U_{lim}}{U_{imax} - U_{imin}} \right) \quad (3)$$

式中: $f(t)$  为  $t$  时刻目标函数; $g_{ij}$  为节点  $i, j$  之间的电导; $N_l$  为线路集合; $N_{PQ}$  为所有 PQ 节点的集合;

$U_i, U_{iN}, U_{lim}, U_{imax}, U_{imin}$  分别为节点  $i$  的电压幅值、额定电压值、电压越限阈值、电压上限值、电压下限值; $\theta_{ij}$  为节点  $i, j$  之间的电压相角差; $\lambda$  为电压越限的惩罚因子,通常取值较大,本文取为 1 000。

那么 1 d 的网络损失量最小的目标函数为

$$F = \min \sum_{i=1}^{24} P_{loss}(t) \quad (4)$$

分布式电源不仅能够给电源提供有功功率,而且能够给电网提供无功功率进行电压支撑,调节无功潮流分布,降低网络损失。无功出力为

$$Q_{max}(t) = \sqrt{S_{DG}^2 - P_{out}^2(t)} \quad (5)$$

式中: $Q_{max}(t)$  为  $t$  时刻分布式电源最大无功功率; $S_{DG}$  为分布式电源额定视在功率; $P_{out}(t)$  为  $t$  时刻分布式电源输出有功功率。

储能装置根据峰时间段放电、谷时间段充电的原则进行削峰填谷,充放电的同时可以对无功功率进行吸收或释放,能够保证配电网安全稳定运行。储能装置的无功输出为

$$Q_{ESSmax}(t) = \sqrt{S_{ESS}^2 - P_{ESSout}^2(t)} \quad (6)$$

式中: $Q_{ESSmax}(t)$  为  $t$  时刻储能装置发出或者吸收的最大无功功率; $S_{ESS}$  为储能装置额定视在功率; $P_{ESSout}(t)$  为  $t$  时刻储能装置输出有功功率的吸收值或者发出值。

### 1.3 约束条件

#### ①潮流约束

$$\begin{cases} P_i = \sum_{i \in N_{ESS}} P_{ESSi} + \sum_{i \in N_{pv}} P_{pvi} + \sum_{i \in N_{wt}} P_{wti} - \sum_{i \in N_{loadi}} P_{loadi} \\ Q_i = \sum_{i \in N_{ESS}} Q_{ESSi} + \sum_{i \in N_{pv}} Q_{pvi} + \sum_{i \in N_{wt}} Q_{wti} + \sum_{i \in N_c} Q_{ci} - \sum_{i \in N_{loadi}} Q_{loadi} \end{cases} \quad (7)$$

式中: $P_i, Q_i$  分别为节点  $i$  有功、无功出力; $P_{ESSi}, P_{pvi}, P_{wti}$  分别为节点  $i$  的储能装置、光伏发电、风力发电的有功出力; $Q_{ESSi}, Q_{pvi}, Q_{wti}, Q_{ci}$  分别为节点  $i$  的储能装置、光伏发电、风力发电、无功调压设备的无功出力; $P_{loadi}, Q_{loadi}$  分别为节点  $i$  的负荷有功、无功功率; $N_{ESS}, N_{pv}, N_{wt}, N_c, N_{loadi}$  分别为储能装置、光伏发电、风力发电、无功调压设备、负荷的节点集合。

#### ②电压约束

$$U_{imin} < U_{ij} < U_{imax} \quad (8)$$

式中: $U_{\min}, U_{\max}$  分别为节点  $i$  的最小、最大电压值。

电压越限阈值设置如下:

$$U_{ij} \begin{cases} U_{\max}, U_i \geq U_{\max} \\ U_{ij}, U_{\min} < U_i < U_{\max} \\ U_{\min}, U_i \leq U_{\min} \end{cases} \quad (9)$$

③电流约束

$$I_{ij} < I_j < I_{j\max} \quad (10)$$

式中: $I_{j\min}, I_{j\max}$  分别为流过节点  $i, j$  的最小、最大电流值。

④储能装置无功出力约束

$$Q_{\text{ESS}\min} < Q_{\text{ESS}} < Q_{\text{ESS}\max} \quad (11)$$

式中: $Q_{\text{ESS}\min}, Q_{\text{ESS}\max}$  分别为节点  $i$  的储能无功输出最小值、最大值。

⑤分布式电源无功出力约束

$$Q_{\text{DG}\min} < Q_{\text{DG}} < Q_{\text{DG}\max} \quad (12)$$

式中: $Q_{\text{DG}\min}, Q_{\text{DG}\max}$  分别为节点  $i$  的分布式电源无功输出最小值、最大值。

⑥无功补偿设备出力约束

$$Q_{\text{cim}\min} < Q_{\text{ci}} < Q_{\text{cim}\max} \quad (13)$$

式中: $Q_{\text{cim}\min}, Q_{\text{cim}\max}$  分别为节点  $i$  的无功补偿设备输出最小值、最大值。

⑦电容器投切数量约束

$$C_{i\min} < C_i < C_{i\max} \quad (14)$$

式中: $C_{i\min}, C_{i\max}$  分别为节点  $i$  的电容器投切数量最小值、最大值。

⑧有载调压变压器档位约束

$$T_{i\min} < T_i < T_{i\max} \quad (15)$$

式中: $T_{i\min}, T_{i\max}$  分别为节点  $i$  的有载调压变压器档位最小值、最大值。

## 2 卷积循环神经网络

### 2.1 卷积神经网络

卷积神经网络是一种深度前馈神经网络,具有很强的特征学习能力。图2为卷积神经网络的结构,主要包含卷积、激活、池化、softmax 4个部分,其中卷积、激活、池化堆叠交替出现。

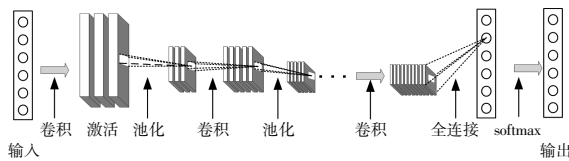


图2 卷积神经网络结构

Fig.2 Convolutional neural network structure

#### 2.1.1 卷积

卷积运算是整个网络的核心,使用卷积核遍历输入的每个点,权值共享,减小网络的训练时间。输入特征的地方与卷积核对应区域进行相乘与求和操作,再加上偏置,形成一个输出特征值。计算式如下:

$$x_{i+1} = W_i \otimes x_i + b_i \quad (16)$$

式中: $x_i$  为前一层的输入特征; $x_{i+1}$  为卷积之后的一层输出特征; $\otimes$  为卷积运算; $W_i$  为卷积核的权重; $b_i$  为偏置。

#### 2.1.2 激活

激活层使用激活函数将卷积后的特征进行非线性变换,常见的激活函数有 Relu, Sigmoid 激活函数。式(16)中添加激活函数后如下所示:

$$y_i = f(x_{i+1}) = f(W_i \otimes x_i + b_i) \quad (17)$$

式中: $y_i$  为经过激活函数后的输出; $f(*)$  为激活函数。

#### 2.1.3 池化

经过卷积处理后输出的特征图个数和维度大大增加了,容易出现维度灾难以及过拟合现象。为了防止上述现象的发生,使用池化可以在保持原来特征的前提下进行降维。常见的池化操作为最大池化,即从池化区域内选出最大值作为该池化区域的代表值,表达式如下:

$$\begin{aligned} \max\text{-pooling}(f[i-1], f[i], f[i+1]) = \\ \max(f[i-1], f[i], f[i+1]) \end{aligned} \quad (18)$$

式中: $\max\text{-pooling}$  为最大池化操作; $f[i]$  为第  $i$  个特征值; $f[i+1]$  为第  $i+1$  个特征值; $f[i-1]$  为第  $i-1$  个特征值; $\max(*)$  为取最大值函数。

#### 2.1.4 全连接

全连接层是卷积神经网络输出和 softmax 分类器的中间层,经过展开平函数变成一维特征,再输出为多分类的向量,层与层之间神经元互相连接起来,如下所示:

$$o(x) = f(wx + b) \quad (19)$$

式中: $o(x)$  为全连接的输出; $x$  为全连接的输入; $f(*)$  为全连接的激活函数; $w$  为全连接的权重; $b$  为全连接的偏置。

卷积神经网络反向传播主要采用梯度下降算法,利用期望输出和训练的样本进行网络参数的估计,主要包括卷积核权重  $W_i$ 、偏置  $b$ 、全连接层权重  $w$ 。根据样本的误差函数即损失函数来计算所有层的误差,进而得出网络的学习方式,不断更

新参数,使实际输出接近期望输出,损失函数的计算式如下:

$$J(W_i, b, w) = \frac{1}{2N} \sum_{i=1}^N \|t_i - y_i\|^2 \quad (20)$$

式中:  $N$  为样本个数;  $t_i$  为期望输出;  $y_i$  为网络实际输出。

使用偏导链式规则,一层一层反向向前传播,求得所有层的灵敏度,计算式如下:

$$\delta_i = \frac{\partial J}{\partial x_i} \quad (21)$$

利用所有层的灵敏度计算出网络损失函数对各个参数的偏导,以及损失函数对  $W_i, b, w$  的偏导,不断更新网络参数。第  $i$  层的卷积层参数更新为

$$W_i^{t+1} = W_i^t - \eta \frac{\partial J}{\partial W_i} \quad (22)$$

$$b_i^{t+1} = b_i^t - \eta \frac{\partial J}{\partial b_i} \quad (23)$$

第  $m$  层的全连接层参数更新为

$$w_m^{t+1} = w_m^t - \eta \frac{\partial J}{\partial w_m} \quad (24)$$

$$b_m^{t+1} = b_m^t - \eta \frac{\partial J}{\partial b_m} \quad (25)$$

式中:  $t$  为迭代次数;  $\eta$  为学习率,可以决定梯度下降算法的步长,  $\eta$  如果设置太小,则会导致学习速度太慢,如果设置太大,则会导致网络发散。

### 2.2 自适应学习率

损失函数是一个待优化的函数,其图像如图 3 所示。图中  $\theta$  为待优化的参数,即  $W_i, b, w$ 。

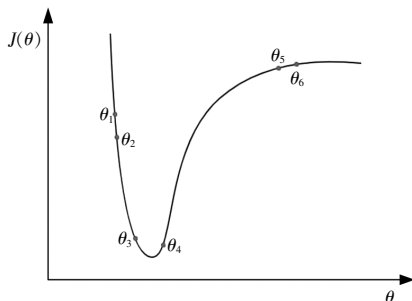


图 3 损失函数图像

Fig.3 Loss function image

由图 3 中可知:  $\theta_3$  和  $\theta_4$  靠近最优解,  $\eta$  应该小一点,防止算法在最优解处发生震荡;  $\theta_1, \theta_2, \theta_5, \theta_6$

远离最优解,  $\eta$  取值应该大一点,可以加快算法的迭代速度,使得算法迅速到达最优点附近。最优点周围的梯度差值  $|g(\theta_4) - g(\theta_3)|$  要高于远离最优点的梯度差值  $|g(\theta_6) - g(\theta_5)|, |g(\theta_2) - g(\theta_1)|$ ,故使用梯度差值能够控制  $\eta$  的大小。即远离最优点时,梯度差值较小,  $\eta$  较大;靠近最优点时,梯度差值较大,  $\eta$  较小。迭代过程中的梯度差计算式如下所示:

$$\Delta g = |g^t - g^{t-1}| \quad (26)$$

式中:  $\Delta g$  为当前迭代和上一次迭代的梯度差;  $g^t$  为第  $t$  次迭代的梯度。

使用梯度差能够求出待优化的损失函数曲率变化,曲率变化自动调整  $\eta$ ,进而自适应调整  $\eta$ 。

利用梯度差进行  $\eta$  更新,将对数函数作为求解的损失函数的复合函数。损失函数不小于零,因此复合函数  $h(\theta^t) = \ln(J(\theta^t) + 1)$  也是一个不小于零的函数。对复合函数求导得出梯度  $\nabla h(\theta^t)$ ,引入动量加速度可以得到复合函数的泰勒展开公式。整个算法的更新方式是通过现在梯度的近似值  $m^t e^t$  和实际值  $g^t$  的不同权值相加代替之前梯度下降算法的梯度,并且利用梯度差实现自适应学习率的网络算法的优化。整个过程计算如下:

$$\begin{cases} h(\theta^t) = \ln(J(\theta^t) + 1) \\ \nabla J(\theta^t) = \nabla h(\theta^t) e^{h(\theta^t)} \\ \nabla h(\theta^t) = \frac{\nabla J(\theta^t)}{J(\theta^t) + 1} \\ m^t = \beta_1 m^{t-1} + (1 - \beta_1) \nabla h(\theta^t) \\ v^t = \beta_2 v^{t-1} + (1 - \beta_2) [\nabla h(\theta^t)]^2 \\ h(\theta^t) \approx v^t = v^{t-1} + m^t (\theta^t - \theta^{t-1}) \\ \theta^{t+1} = \left[ \theta^t - \eta \frac{\beta_2 g^t + (1 - \beta_2) m^t e^t}{\Delta g + \varepsilon} \right] \end{cases} \quad (27)$$

式中:  $\theta^t$  为第  $t$  次迭代时待优化参数;  $m^t, v^t$  分别为复合函数梯度和梯度平方的指数平均加权移动值;  $\beta_1$  为  $m^t$  的衰减系数,取值为 0.9;  $\beta_2$  为当前近似梯度值和实际值结合的权重值,取值为 0.3;  $\varepsilon$  为调节参数,防止梯度差的分子为零,取值为 0.01。

### 3 无功协调优化方案

通过粒子群优化算法求解以最小网络损耗和最低电压偏移为目标的配电网源网荷储无功优化模型,求解流程如图 4 所示。

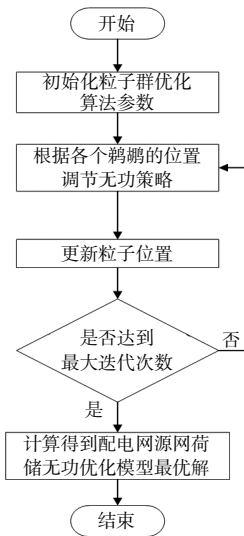


图4 模型求解流程

Fig.4 Model solving process

模型计算过程如下:

步骤 1:初始化粒子群优化算法参数:粒子位置、速度、迭代次数、种群规模;

步骤 2:根据各个粒子的位置调节电容器投切数量、无功补偿容量、有载调压变压器档位、储能装置充放电策略;

步骤 3:更新粒子位置,判断是否达到最大迭代次数,如果没有,则返回步骤 2;如果达到最大迭代次数,则转到下一步;

步骤 4:计算得到配电网源网荷储无功优化模型最优解。

配电网源网荷储无功协调优化反映了电网运行场景和无功调压设备出力、储能装置充放电策略之间的隐藏关系,使用卷积神经网络构建电网运行场景和无功调压设备出力、储能装置充放电策略的映射关系。基于卷积神经网络的配电网源网荷储无功协调优化模型,将卷积神经网络的一次卷积、池化看成一个隐藏层,配电网无功优化模型如图 5 所示。

基于自适应学习率卷积神经网络的配电网源网荷储无功协调优化模型通过训练集确定网络结构,网络的输入为电网场景参数:光伏发电出力、风力发电出力、负荷功率;网络的输出为无功协调优化策略:电容器投切数量、无功补偿容量、有载调压变压器档位、储能装置充放电策略。无功优化流程图如图 6 所示。

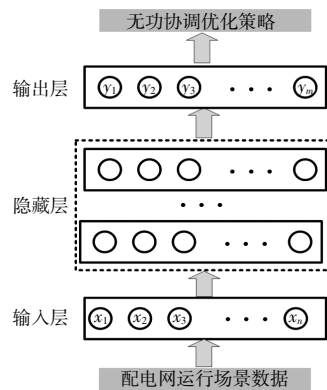


图5 无功协调优化模型

Fig.5 Reactive power coordination optimization model

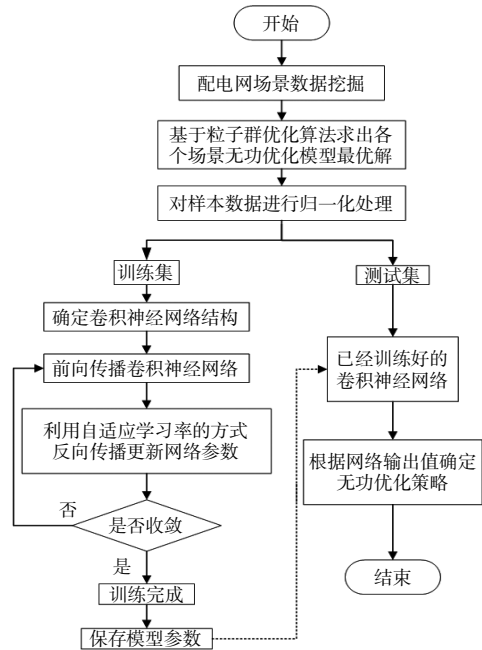


图6 无功优化流程

Fig.6 Reactive power optimization process

无功优化流程步骤如下。

步骤 1:数据预处理。使用蒙特卡洛方法生成网络结构、分布式电源数据以及负荷容量的配电网场景,基于粒子群优化算法求出各个场景的配电网源网荷储无功优化模型最优解,形成各个场景的最佳无功协调优化策略。生成无功协调优化训练样本,将样本分成训练集和测试集。输入数据为不同维度的配电网数据,对数据进行归一化处理:

$$\hat{x}_i = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} \quad (28)$$

式中: \$\hat{x}\_i\$ 为归一化后样本数据; \$x\_i\$ 为原始数据;

$\min(x_i), \max(x_i)$  分别为最小值和最大值函数。

步骤 2: 卷积神经网络模型确定。确定网络层数和每层神经元个数, 本文使用 Relu 激活函数, 能够使得一些神经元值为 0, 防止出现过拟合的现象。

步骤 3: 训练集网络参数确定。使用训练集的电网场景作为网络输入, 前向传播卷积神经网络, 通过网络的输出无功优化策略和实际优化策略得出预测值和实际值的误差损失函数。根据损失函数最小原则, 采用自适应学习率的方式反向更新网络参数, 最后形成基于卷积神经网络的配电网源网荷储无功协调优化模型。

步骤 4: 测试集无功优化策略形成。利用训练好的卷积神经网络得出测试集下的配电网无功优化策略, 验证本文方法的有效性。

#### 4 仿真验证

##### 4.1 数据集构建

为了验证本文方案的有效性, 通过动模试验模拟光伏发电系统、风力发电系统、储能系统输出, 搭建了如图 7 所示的 IEEE33 节点配电网模型。

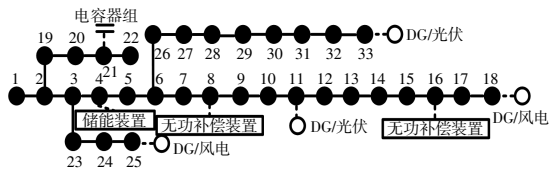


图 7 配电网模型

Fig.7 Distribution network model

分布式电源的输出数据如图 8 所示。有载调压变压器变比步长为 0.012 5; 风力发电接入节点为 18, 25, 额定容量为 500 kW; 光伏发电接入节点为 11, 33, 额定容量为 500 kW; 无功补偿装置接入节点为 8, 16, 额定容量为 400 kVar; 电容器

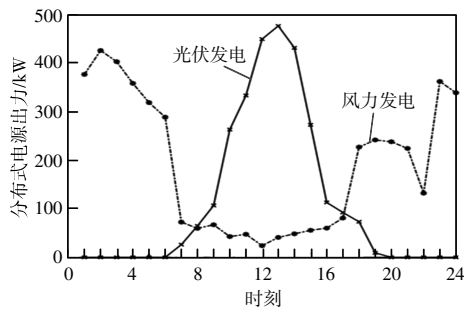


图 8 某天分布式电源出力

Fig.8 Distributed power generation output on a certain day

组接入节点为 21, 容量为 200 kVar; 储能装置接入节点为 4, 总容量为 4 000 kW·h, 运行年限为 10 a, 最大充电和放电次数为 9 000 次, 额定功率为 400 kW。

##### 4.2 网络结构

基于自适应学习率卷积神经网络的配电网源网荷储无功协调优化方法的关键是构建表征电网运行特性的输入、输出样本集。网络的输入为电网场景参数: 2 个光伏发电机出力、2 个风力发电机出力、33 个节点负荷功率; 网络的输出为无功协调优化策略: 1 个电容器投切数量、2 个无功补偿器容量、1 个有载调压变压器档位、1 个储能装置充放电策略。因此, 网络的输入数据维度为 37×1, 输出维度为 5×1。

网络训练的效果用优化策略偏差率  $\delta$  表示, 含义为网络输出的无功优化策略和实际策略之间的偏差,  $\delta$  值越小, 说明网络无功优化策略越好。

$$\delta = \frac{1}{K} \sum_{i=1}^K \left( \frac{c_i}{C_i} + \frac{P_{ESS}}{P_{ESS}} + \frac{q_{SVG}}{Q_{SVG}} + \frac{t_i}{T_i} \right) \quad (29)$$

式中:  $C_i, c_i$  分别为第  $i$  个电容器的实际投切数和网络输出投切数;  $P_{ESS}, p_{ESS}$  分别为第  $i$  个储能装置的实际充放电量和网络输出充放电容量;  $Q_{SVG}, q_{SVG}$  分别为第  $i$  个无功补偿器的实际补偿容量和网络输出补偿容量;  $T_i, t_i$  分别为第  $i$  个有载调压变压器的实际档位和网络输出档位;  $K$  为无功优化节点数。

确定卷积神经网络各层神经元个数: 设置卷积神经网络隐藏层个数为 1, 将隐藏层神经元个数从 50 设置到 300; 使用 100 组数据集进行训练, 观察策略  $\delta$  的变化, 以  $\delta$  最低原则确定第一层隐藏层神经元个数; 固定第一层隐藏层神经元个数, 将第二层隐藏层神经元个数从 50 设置到 300, 观察策略  $\delta$  的变化, 以  $\delta$  最低原则确定第二层隐藏层神经元个数; 依次类推, 当  $\delta$  变大时, 停止增加隐藏层, 得出网络结构。

通过实验结果, 得到如图 9 所示的策略  $\delta$  和隐藏层层数、隐藏层神经元个数之间的关系。由图中可知, 当网络只有 1 层隐藏层, 神经元个数为 200 时, 策略  $\delta$  最小, 因此选择第一层隐藏层神经元个数为 200。继续增加第二层隐藏层神经元个数, 当第二层隐藏层神经元个数为 150 时, 策略  $\delta$

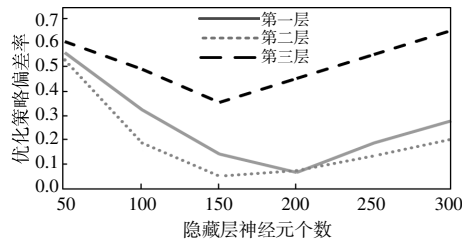


图9 无功优化策略偏差率和隐藏层的层数、神经元个数之间的关系

Fig.9 Relationship between the deviation rate of reactive power optimization strategy and the number of hidden layers and neurons in the hidden layer

最小,因此选择第二层隐藏层神经元个数为150。但是,当第三层隐藏层神经元个数增加时,策略 $\delta$ 较大,网络出现过拟合现象,效果较差。因此本文网络隐藏层个数为2,结合输入、输出,网络的结构设置为37-200-150-5。

### 4.3 结果分析

为了验证网络的训练效果,做出网络损失函数和迭代次数之间的关系曲线,见图10。由图中可知,当迭代次数达到200时,网络趋于稳定,最小损失函数值达到0.32,说明网络训练效果较好。

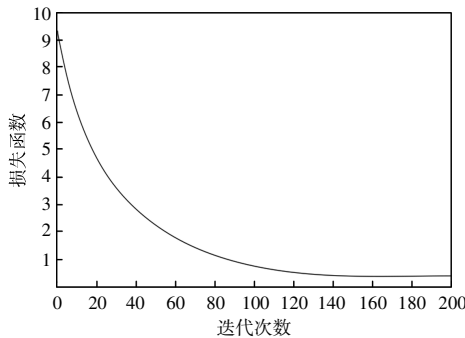


图10 损失函数和迭代次数的关系

Fig.10 Relationship between loss function and iteration times

为了验证本文方法的无功优化效果,将本文所提的自适应学习率卷积神经网络和卷积神经网络、支持向量机进行比较,得出不同方法下的网损率 $P_{loss}$ 和电压偏移率 $\Delta U_{ave}$ 。

$$\begin{cases} P_{loss} = \frac{P_T - P_Y}{P_Y} \\ \Delta U_{ave} = \frac{1}{n} \sum_{i=1}^n \frac{|U_{i,E} - U_{i,Y}|}{U_{i,Y}} \end{cases} \quad (30)$$

式中: $P_T$ 为实际网损; $P_Y$ 为优化方法的网损; $U_{i,E}$ 、 $U_{i,Y}$ 分别为第*i*个节点的额定电压、实际电压。

不同方法的网损率和电压偏移率结果如图11,12所示。由图中可知,本文所提自适应学习率

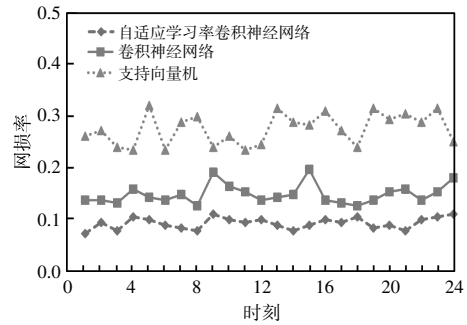


图11 不同方法网损率对比

Fig.11 Comparison of network loss rates using different methods

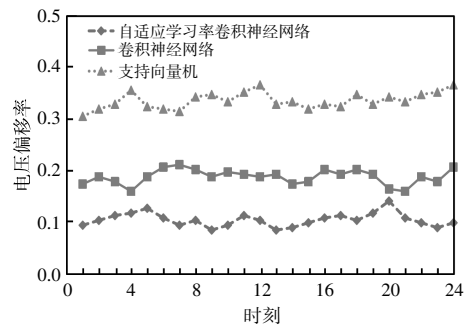


图12 不同方法电压偏移率对比

Fig.12 Comparison of voltage offset rates different methods

卷积神经网络方法的网损率、电压偏移率最低,验证了本文方法对配电网无功优化的效果最好。

为了验证本文方法对于不同场景均有较好的效果,对一年四季不同风力、光照的电网场景进行验证。每种场景选择训练集个数为400,测试集个数为100,不同方法的不同场景下的网损率和电压偏移率如表1所示。

表1 不同场景下的结果对比

Table 1 Comparison of results in different scenarios

场景	网损率			电压偏移率		
	自适应学习率卷积神经网络	卷积神经网络	支持向量机	自适应学习率卷积神经网络	卷积神经网络	支持向量机
1	0.008	0.012	0.028	0.012	0.018	0.032
2	0.005	0.013	0.027	0.011	0.019	0.035
3	0.006	0.016	0.025	0.013	0.021	0.037
4	0.009	0.018	0.029	0.014	0.023	0.035
5	0.012	0.017	0.031	0.011	0.018	0.038
6	0.011	0.021	0.028	0.012	0.017	0.039
7	0.008	0.014	0.027	0.013	0.016	0.034
8	0.009	0.014	0.026	0.009	0.017	0.037
9	0.007	0.016	0.027	0.011	0.019	0.038
10	0.013	0.015	0.029	0.012	0.018	0.035

由表 1 可知,本文方法在不同场景的网损率、电压偏移率均最低,验证了本文方法的有效性和实用性。

## 5 结语

本文提出了一种基于自适应学习率卷积神经网络的配电网源网荷储无功协调优化技术,构建了无功优化模型,通过卷积神经网络特征映射电网运行场景和无功调压设备、储能充放电策略之间的关系。同时,利用自适应学习率更新参数,提高模型训练效率,输出电力系统无功电压主动优化控制策略,实现新型配电网源网荷储无功协调优化调节。

## 参考文献:

- [1] 赵静波,卫志农,刘建坤,等.电力系统线性化动态最优潮流模型[J].电力系统自动化,2018,42(20):86-92.
- [2] 余娟,颜伟,徐国禹,等.基于预测-校正原对偶内点法的无功优化新模型[J].中国电机工程学报,2005,25(11):146-151.
- [3] 罗铸.电力系统无功和电压的优化调节[J].电力系统自动化,1988,12(1):9-15.
- [4] 徐建亭,王秀英,李兴源.电力系统电压无功的序列二次规划算法[J].电力系统自动化,2001,25(23):4-8.
- [5] 郭清元,吴杰康,莫超,等.基于混合整数二阶锥规划的新能源配电网电压无功协同优化模型[J].中国电机工程学报,2018,38(5):1385-1396.
- [6] 赖晓文,马晓伟,白杨,等.基于混合整数二阶锥规划的动态无功优化方法[J].电力系统自动化,2017,41(17):37-42.
- [7] 于琳,孙莹,徐然,等.改进粒子群优化算法及其在电网无功分区中的应用[J].电力系统自动化,2017,41(3):89-95,128.
- [8] 李晓利,高金峰.用于配电网多目标无功优化的改进粒子群优化算法[J].电力自动化设备,2019,39(1):106-111.
- [9] 李军,陈利,李华东,等.基于遗传算法的多机型多热源复杂热电负荷优化分配方法[J].节能技术,2021,39(2):185-188,192.
- [10] 黄俊辉,汪惟源,王海潜,等.基于模拟退火遗传算法的交直流系统无功优化与电压控制研究[J].电力系统保护与控制,2016,44(10):37-43.
- [11] 侯学勇,丁晓群.基于遗传-模拟退火算法的配电网电容器优化配置[J].电网技术,2005,29(4):81-84.
- [12] 刘科研,盛万兴,李运华.基于改进遗传模拟退火算法的无功优化[J].电网技术,2007,31(3):13-18.
- [13] 张森,陈栩杰,康家熙,等.考虑电动汽车与共享私人车位交易的微电网需求侧管理[J].智慧电力,2020,48(5):34-40,46.
- [14] 周鑫,许守东,段晓,等.基于聚类蚁群算法的动态无功优化[J].云南电力技术,2014,42(5):40-46.
- [15] Chen Xu, Yi Yingqi, Zhang Yongjun, et al. Approach to setting gateway reactive power control band for distribution networks with wind power [J]. IET Generation, Transmission & Distribution, 2017, 11(3): 596-604.
- [16] 刘科研,季玉琦,陆凌芝,等.基于负荷分布匹配与熵权法的配电网无功优化[J].电网技术,2017,41(12):3980-3988.
- [17] 邵美阳,吴俊勇,石琛,等.基于数据驱动和深度置信网络的配电网无功优化[J].电网技术,2019,43(6):1874-1885.
- [18] 安然,吴俊勇,石琛,等.基于随机矩阵和历史场景匹配的配电网无功优化[J].中国电力,2020,53(4):69-78.
- [19] 黄景林,彭显刚,简胜超,等.基于深度学习与不平衡样本集的输电线路故障分类[J].智慧电力,2021,49(2):114-119.
- [20] 万杰,陈宁,钱敏慧,等.基于混合型深信度网络的风速日前预测研究[J].节能技术,2016,34(1):81-86.
- [21] Abdeljaber O, Avic O, Kiranyaz S, et al. Real-time vibration-based structural damage detection using one dimensional convolutional neural networks [J]. Journal of Sound and Vibration, 2017, 388: 154-170.

## Reactive power coordination optimization technology for source–network–load–storage in new distribution network based on adaptive learning rate convolutional neural network

Qian Jinbao<sup>1</sup>, Liu Xiaoguang<sup>2</sup>, Cai Xi<sup>2</sup>, Liu Yi<sup>2</sup>, Dai Jianfeng<sup>3</sup>

(1.Electric Power Research Institute of State Grid Hubei Electric Power Co.,Ltd., Lanzhou 730000, China; 2.Gansu Tongxing Intelligent Technology Development Co.,Ltd., Lanzhou 730050, China; 3.College of Automation & College of Artificial Intelligence, Nanjing University of Posts and Telecommunications, Nanjing 210023, China)

**Abstract:** With the promotion of the "dual carbon" goal, the capacity of distributed new energy connected to the power grid has significantly increased. The use of distribution network source network load storage coordination optimization strategy is an important method to achieve distributed new energy consumption, among which reactive power optimization can ensure the safe and stable operation of the power grid. This article proposes an adaptive learning rate convolutional neural network based optimization technique for load storage and reactive power coordination in distribution networks. Firstly, a reactive power optimization model is constructed with the goal of minimizing network loss and voltage offset. Secondly, utilizing the powerful nonlinear fitting ability of convolutional neural networks, the mapping relationship between power grid operation scenarios, reactive power regulation equipment, and energy storage charging and discharging strategies is excavated. Adaptive learning rate is introduced to update network parameters and improve network training efficiency. Finally, by controlling the charging and discharging conditions of reactive power regulation equipment and energy storage devices to coordinate the output of distributed power sources, active optimization control of reactive power and voltage in new distribution network is achieved. After simulation verification of the IEEE33 node power grid model, the results show that the proposed optimization method for load storage and reactive power coordination in the distribution network source network improves the voltage regulation ability of the power system, laying a good foundation for the safe and reliable operation of the distribution network.

**Keywords:** distributed new energy; optimization of source network load storage coordination; reactive power optimization; adaptive learning rate; convolutional neural network