Explainable AI in Deep Reinforcement Learning Models for Power System Emergency Control

Ke Zhang, *Student Member, IEEE*, Jun Zhang, *Senior Member, IEEE*, Pei-Dong Xu, *Graduate Student Member, IEEE*, Tianlu Gao, *Member, IEEE*, and David Wenzhong Gao, *Fellow, IEEE*

***Abstract*— Artificial intelligence (AI) technology has become an important trend to support the analysis and control of complex and time-varying power systems. Although deep reinforcement learning (DRL) has been utilized in the power system field, most of these DRL models are regarded as black boxes, which are difficult to explain and cannot be used on occasions when human operators need to participate. Using the explainable AI (XAI) technology to explain why power system models make certain decisions is as important as the accuracy of the decisions themselves because it ensures trust and transparency in the model decision-making process. The interpretability issue in DRL models in power system emergency control is discussed in this article. The proposed interpretable method is a backpropagation deep explainer based on Shapley additive explanations (SHAPs), which is named the Deep-SHAP method. The Deep-SHAP method is adopted to provide a reasonable interpretable model for a DRL-based emergency control application. For the DRL model, the importance of input features has been quantified to obtain contributions for the outcome of the model. Further, feature classification of the inputs and probabilistic analysis of the outputs in the XAI model is added to interpretability results for better clarity.**

***Index Terms*— Deep reinforcement learning (DRL), emergency control, explainable artificial intelligence (XAI), power system.**

NOMENCLATURE

$\mathbf{x}_i^{\text{baseline}}$	The reference value of the features matrix.
a_k^h	Output actions.
h	The category of the output actions.
K	The total number of interaction steps in training for the DRL model.
l	A positive integer.
M	The subset union M of all the features.
N_A	The number of output neurons of the neural network.
N_{h1}	The first hidden layers in the DQN network.
N_{h2}	The second hidden layers in the DQN network.
N_P	The number of loads related features.
N_r	The total time step.
N_S	The number of input neurons in the model.
N_V	The number of voltage magnitudes related features.
$P_{S(x_i)}$	The normalized exponential function.
$S(x_i)$	The SHAP value of feature x_i.
x_i	The ith feature value.
$\mathcal{R}_{\mathbf{A}}$	The output datasets.
$\mathcal{R}_{\mathbf{S}}$	The input datasets.
$F_{\text{diff}}(x_i)$	The difference between with and without x_i.
x_i^k	The feature x_i in the kth interaction steps in training.
$\mathbf{A}_k$	The output matrix at the kth interaction steps in training for the DRL model.
$\mathbf{O}_t$	The observed grid state.
$\mathbf{S}_k$	The input matrix at the kth interaction steps in training for the DRL model.
$\mathbf{X_F}$	The vector set of input features.
$\mathbf{V}_m(t)$	The bus voltage magnitude matrix for bus m.
$\mathbf{P}_n$	The load shedding amount in p.u. for load bus n at time step t for observed buses.
$C_{\Delta \mathbf{x}_i^k \Delta a_k^h}$	The marginal contribution value of feature x_i to the output action a_k^h.

Manuscript received January 9, 2021; revised May 16, 2021; accepted July 3, 2021. Date of publication August 4, 2021; date of current version April 1, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0101504 and in part by the Science and Technology Project of the State Grid Corporation of China (SGCC): Fundamental Theory of Human-in-the-loop Hybrid-Augmented Intelligence for Power Grid Dispatch and Control. *(Corresponding author: Jun Zhang.)*

Ke Zhang, Jun Zhang, Pei-Dong Xu, and Tianlu Gao are with the School of Electrical Engineering and Automation, Wuhan University, Wuhan 430072, China (e-mail: zhangke@whu.edu.cn; jun.zhang.ee@whu.edu.cn; xupd@whu.edu.cn; tianlu.gao@whu.edu.cn).

David Wenzhong Gao is with the Department of Electrical and Computer Engineering, University of Denver, Denver, CO 80208 USA (e-mail: wenzhong.gao@du.edu).

Digital Object Identifier 10.1109/TCSS.2021.3096824

I. INTRODUCTION

RECENT utilization and applications of renewable energies, electric vehicles, power demand response, and energy storage systems cause expansion of capacity and scale of power systems, and also, the issues of uncertainty and complexity in power system operation become more and more prominent.

The power system requires high security and transparency in its analysis and control. Artificial intelligence (AI) technology has become an important means to support the new generation of power systems due to its potential of improving the efficiency, consistency, and accuracy of decision-making. It is worth noting that although AI applications become more and more popular, which often makes direct and critical decisions, lack of interpretability and transparency to human operators is an important factor that prevents its applicability, especially for the deep reinforcement learning (DRL) model.

In the past few years, DRL has been used in different applications for reliable and secure power system operation.

In [1], DRL was used to develop a dynamic load shedding scheme for short-term voltage control. A data-driven cooperative control method based on multiagent DRL for load frequency control of multiarea power systems was reported in [2]. In [3], DRL was applied to determine generation unit tripping in case of emergency circumstances. In [4], the DRL method was adopted to utilize the operation of storage devices in microgrids. In [5], a graph reinforcement learning method was proposed for the real-time dispatch in the power system via topology control.

Most of these DRL models are regarded as black boxes for human operators, which do not reveal their internal working mechanisms. However, in many applications, human operators need to make final decisions by operation policies and regulations and require that AI's recommendations are self-explained. It is difficult to interpret deep neural network-based complex AI models, and in the presence of this challenge, explainable AI (XAI) is proposed and utilized to make AI systems feasible to understand for human operators [6].

Interpretability is always a critical weakness of DRL models, and it is believed that the interpretability technology can help break through this bottleneck, however, few articles investigate the DRL interpretability issue [7], and how to implement the interpretability technology in DRL models is also a necessary and urgent task.

Most of the current interpretable models fall into the types of local and agnostic models. The commonly used agnostic models mainly include visualization methods, knowledge extraction, influence methods, and example-based explanations. To understand machine learning (ML) models, visualization methods [8]–[10] are often adopted, a natural idea is to visualize and explore the contents hidden in neural units. The method of knowledge extraction in an agnostic model is popular at present [11]. Model distillation [12]–[14] in knowledge extraction is a hot topic in the field of AI for complex models. Meanwhile, more and more scholars propose to use knowledge graph [15], [16], as a means of knowledge extraction, to achieve interpretability in ML. The influence method is widely used in the agnostic models, which estimates the importance or relevance of features by changing inputs or internal components and documenting the impact of changes on model performance [17], [18]. Additionally, example-based and mixed explanations [19], [20] are also applied in ML interpretability.

The ability to explain why an ML model can make a certain decision in understandable terms becomes very important because it ensures trust and transparency in the decision-making process. For deep learning models, some strategies have been used for interpretability [21]. A visual analysis framework of interactive and interpretable deep learning model was proposed in [22]. In [23], an interpretable model for deep learning models is proposed to assign importance scores for the input features. However, for DRL models, few articles mentioned how to construct an appropriate interpretable model.

The Deep-SHAP method proposed in this article is a deep explainer based on the SHAP method to implement the interpretability of the DRL model. The SHAP method [24] is based on feature importance, which quantifies the contribution of each input variable, or feature, to the functionality of a complex ML model. To measure the importance of features, the increase of model error is calculated by replacing each corresponding feature. At present, the SHAP method has been used in some cases to calculate the importance of features in ML models to obtain an intuitive explanation. The discovery of important biomarkers in [25] is helpful for accurate diagnosis and prediction of certain cancer types, which uses gradient enhancement trees and the SHAP method. In [26], a tree explainer based on the SHAP method is a tool for explaining the global model structure based on local interpretations. The interpretation ability of Shapley values is used to assign a specific predicted importance value to each feature [27]. In [28], the SHAP method was applied in solar photovoltaic forecasting models to provide improvements and point out relevant parameters.

To verify the performance of the proposed approach for DRL models, an adaptive emergency control scheme of power systems based on DRL is utilized [29], [30]. Then, the Deep-SHAP method is applied to the undervoltage load shedding of power systems based on the DRL model.

The interpretability issue in the DRL model for power system control is discussed in this article. The application of the Deep-SHAP method in the DRL model of power system emergency control is implemented and analyzed. Based on the original SHAP method, feature classification of the inputs and probabilistic analysis of the outputs in the XAI model are added to interpretability results for better clarity. Importantly, understanding the internal mechanism of the proposed DRL model through the Deep-SHAP method assists human operators to make more accurate decisions and deployments.

The rest of this article is organized as follows. Section II describes the power grid problem and DRL scheme for developing and bench-marking DRL algorithms. Section III details the proposed Deep-SHAP method. The model interpretations are shown in Section IV. The conclusions and future work are provided in Section V.

II. Emergency Control Model in Power System

Power system emergency control is the final defense line to ensure the safety [31]. A DRL model in grid control is adopted to verify the performance of the Deep-SHAP method in interpretability, which is shown in Fig. 1. The experimental scenario is set in the modified IEEE 39-bus system with the buses of 4, 7, and 18 being heavy-load areas, and step-down transformers are added to these load buses. The causes of the fault are considered as stalling and tripping of major loads, such as stalling of residential air-conditioner motors and prolonged tripping, in the grid. After the faults are cleared, the safety standard requires that voltages should return to at least 0.8, 0.9, and 0.95 p.u. within 0.33, 0.5, and 1.5 s, respectively [29]. Fault-induced delayed voltage recovery refers to the phenomenon that the system voltage remains at a significantly reduced level for several seconds after the fault is cleared [30]. Under such circumstances,

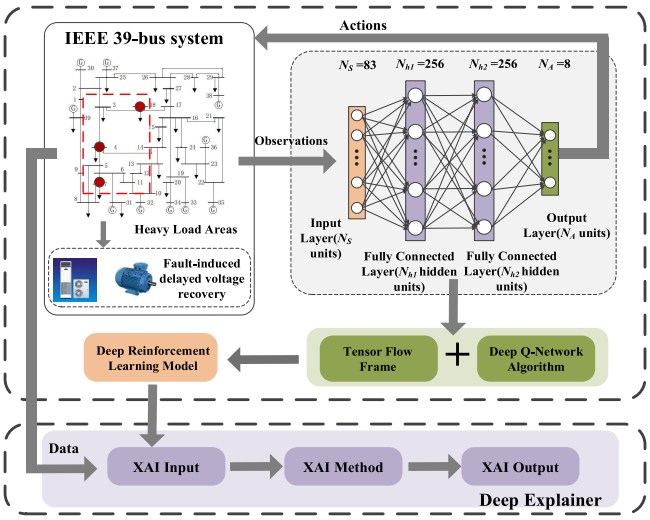


Fig. 1. XAI framework for power grid DRL.

low-voltage load shedding was introduced to solve the problem of fault-induced delayed voltage recovery.

The deep Q-network (DQN) algorithm was applied for developing a coordinated undervoltage load shedding scheme. The neural network was designed with a topology composed of four layers according to the diagram in Fig. 1, which is a fully connected network. N_i corresponds to the input layer of the neural network, including 83 observed features. The basis of feature selection is prior knowledge and feature engineering. First of all, the range of features is narrowed through the preliminary screening of features. Furthermore, the original data is processed by feature engineering. The observations include the voltage magnitudes of buses 4, 7, 8, and 18 and the low-voltage sides of the step-down transformers connected with them in ten observation times. Moreover, the percentage of a load of buses 4, 7, and 18 at the current time is also used as observations. The number of input neurons in the model is

$$N_S = (N_r - 1)N_V + (N_V + N_P) \tag{1}$$

where N_V is the number of voltage magnitudes related features and N_P is the number of loads related features. The total time step $N_r = 10$ in our model, and the DQN network has two hidden layers N_{h1} and N_{h2}, each with 256 neurons as shown in Fig. 1.

For the aforementioned DRL model in the power grid, the input matrix can be expressed as

$$\mathbf{S}_k = \begin{bmatrix} \mathbf{O}_{t-N_r+1} & \cdots & \mathbf{O}_{t-N_r+l} & \cdots & \mathbf{O}_t \end{bmatrix}^T \tag{2}$$

where $\mathbf{S}_k$ is the input matrix at the kth interaction steps in training for the DRL model, $\mathbf{O}_t$ is the observed grid state, l is a positive integer. The ten latest observation states are stacked and used as inputs for DQN. At time $t - N_r + l$, voltage magnitudes matrix $\mathbf{V}_m(t - N_r + l)$ on the observed bus m, which contains the voltage magnitudes of bus m and low-voltage sides of the step-down transformers connected with it, are included in $\mathbf{O}_{t-N_r+l}$

$$\mathbf{O}_{t-N_r+l} = \mathbf{V}_m(t - N_r + l). \tag{3}$$

There are two parts in the observed states matrix $\mathbf{O}_t$, $\mathbf{V}_m(t)$ is the bus voltage magnitude matrix for bus m, $\mathbf{P}_n$ is the load shedding amount in p.u. for load bus n at time step t for observed buses

$$\mathbf{O}_t = \begin{bmatrix} \mathbf{V}_m(t) & \mathbf{P}_n \end{bmatrix}^T. \tag{4}$$

N_A is the number of output neurons of the neural network, including 8 actions. The control actions for buses 4, 7, and 18 at each action time step include no load shedding and 20% load shedding. The above actions will be used to control the undervoltage load shedding in the IEEE 39-bus system

$$\mathbf{A}_k = \begin{bmatrix} a_k^1 & a_k^2 & \cdots & a_k^h \end{bmatrix}^T \tag{5}$$

where $\mathbf{A}_k$ is the output matrix at the kth interaction steps in training for the DRL model. a_k^h represents output actions, h represents the category of the output actions.

Although the decisions can be made through the DRL model, it is unknown why the model makes them. This opaque model is a great barrier to using advanced neural network-based AI approaches. Under this circumstance, XAI technology is needed to provide proper corresponding explanations. As shown in Fig. 1, both the trained DRL model and data from the IEEE 39-bus system are utilized as the input of the explainer.

III. Proposed Deep-SHAP Method

Interpretability means that a clear summary of specific tasks should be given by certain algorithms and need to connect with the defined principles in human cognition. The goal of XAI is to maintain a high level of predictive accuracy which helps human beings understand, trust, and manage AI models [6]. Under XAI technology, operators can make decisions for ML models in an understandable way and can verify the decisions made by AI. Finally, operators gain confidence in the applicability of AI systems with the explanation of the decision mode [32].

For complex neural network models such as DRL, it cannot be used in critical security-related occasions without explicitly explaining their decision mechanism. The Deep-SHAP method, which belongs to the feature importance method, is proposed in this article to explain the aforementioned DRL model in the power system. The influence of each input feature on a certain output is calculated to obtain the ranking of the importance of the features so that the decision-making process can be explained by the features with different great contributions.

A. Shapley Method

For nonlinear complex models such as the DRL model, the Shapley method can be adopted in cooperative game theory to compute the contribution of every single prediction in the model [20]. In order to compute the contribution of each single input feature in the model, first of all, find the subset union M of all the features $\mathbf{X_F}$ except feature x_i

$$\mathbf{X_F} = \begin{bmatrix} x_1 & \cdots & x_i & \cdots & x_{N_S} \end{bmatrix}^T \tag{6}$$

where x_i is the ith feature value, $\mathbf{X_F}$ is the vector set of input features, $x_i \in \mathbf{X_F}$. For each case in the subset union, the results with and without x_i are calculated. If the feature x_i is not included, the value of x_i is directly replaced by the mean value of all x_i in the datasets $\mathcal{R}_\mathbf{S}$

$$\mathcal{R}_\mathbf{S} = \begin{Bmatrix} \mathbf{S}_1 & \cdots & \mathbf{S}_k & \cdots & \mathbf{S}_K \end{Bmatrix} \tag{7}$$

where K is the total number of interaction steps in training for the DRL model. The difference $F_{\text{diff}}(x_i)$ between with and without x_i is calculated as following to obtain the marginal contribution:

$$F_{\text{diff}}(x_i) = f_x(M \cup \{x_i\}) - f_x(M) \tag{8}$$

where $f_x(M \cup \{x_i\})$ is the outcome from M without feature x_i, $f_x(M)$ is the model outcome from M. The weighted mean value of all the marginal contributions to obtain the feature importance of x_i is computed as following:

$$\text{SHAP}(x_i) = \sum_M \left(\frac{|M|!(k - |M| - 1)!}{k!} \cdot F_{\text{diff}}(x_i) \right) \tag{9}$$

where $(|M|!(k - |M| - 1)!/N_S!)$ is the weight of M, $M \subseteq \{x_1, \ldots, x_{N_S}\} \backslash \{x_i\}$.

The Shapley method ensures the difference between the value and the average outcome value is distributed among features on average. Moreover, it allows for comparative interpretation such as the comparison between single data points. However, the computational complexity of the Shapley method is $O(2^n)$, the Shapley method requires a high computational load as all the feature combinations need to be used. As a result, the computational complexity of the Deep-SHAP method is $O(n)$, the Deep-SHAP method is proposed to improve computational efficiency.

B. Deep-SHAP Method

The XAI algorithm framework based on the Deep-SHAP method is presented in Fig. 2. The inputs of the explainer are the DRL model states, including model inputs, DRL algorithm, and model outputs. The deep explainer consists of two parts, one is the Deep-SHAP layer, the backpropagating strategy is adopted in this part to calculate the importance of input features. The other part is the softmax layer, a probabilistic representation method is proposed in the softmax layer. As for the outputs of the explainer, it consists of a feature analysis layer and a visualization layer. In the feature analysis layer, the classification is based on the different properties of features. The visualization layer presents clear and intuitive interpretation results through various visualization forms. Finally, some tasks for the explainer are presented, including explanation, diagnosis, refinement, and human–computer interaction.

The SHAP values should be estimated by all possible feature alliances with and without feature x_i. However, when the number of features is large, the number of possible alliances will increase exponentially. Therefore, the deep learning of important features (DeepLIFT) method, which is based on a backpropagating strategy, is added in Shapley to obtain a more advanced algorithm. In the DeepLIFT method, important signals, such as the differences of inputs, are transmitted from the outputs $\mathbf{A}_k$ to specific inputs $\mathbf{S}_k$ through a neural network. In our DRL model, combined with (2)–(4), the differences of inputs are expressed as

$$\begin{aligned} \Delta \mathbf{x}_i^k &= x_i^k - \mathbf{x}_i^{\text{baseline}} = \Delta \mathbf{S}_k \\ &= [\, \Delta \mathbf{V}_m(t - N_r + 1) \quad \cdots \\ &\qquad \Delta \mathbf{V}_m(t - N_r + l) \quad \cdots \quad \Delta \mathbf{V}_m(t) \quad \Delta \mathbf{P}_n \,]^T \end{aligned} \tag{10}$$

where x_i^k represents feature x_i in the kth interaction steps in training, $\mathbf{x}_i^{\text{baseline}}$ represents the reference value of the features matrix. $\Delta \mathbf{x}_i^k$ is a vector of $K \times 1$. The marginal contribution value of each feature to the output actions can be expressed as

$$C_{\Delta \mathbf{x}_i^k \Delta a_k^h} = \sum_{k=1}^{K} \left(\nabla f\left(\mathbf{x}_i^k\right) \times \Delta \mathbf{x}_i^k \right) \tag{11}$$

where $C_{\Delta \mathbf{x}_i^k \Delta a_k^h}$ is the marginal contribution value of feature x_i to the output action a_k^h in the datasets $\mathcal{R}_\mathbf{S}$. $\nabla f(\mathbf{x}_i^k)$ is the gradient of a_k^h relative to $\mathbf{x}_i^k$, combined with (5) and (10), can be expressed as

$$\begin{aligned} \nabla f(\mathbf{x}_i^k) &= \frac{\partial \mathbf{A}_k}{\partial \mathbf{x}_i^k} \\ &= \begin{cases} \left[\dfrac{\partial a_k^1}{\partial \mathbf{V}_m(t - N_r + 1)} \cdots \dfrac{\partial a_k^1}{\partial \mathbf{V}_m(t - N_r + l)} \right. \\ \qquad \left. \cdots \dfrac{\partial a_k^1}{\partial \mathbf{V}_m(t)} \ \dfrac{\partial a_k^1}{\partial \mathbf{P}_n} \right]^T \\ \vdots \\ \left[\dfrac{\partial a_k^h}{\partial \mathbf{V}_m(t - N_r + 1)} \cdots \dfrac{\partial a_k^h}{\partial \mathbf{V}_m(t - N_r + l)} \right. \\ \qquad \left. \cdots \dfrac{\partial a_k^h}{\partial \mathbf{V}_m(t)} \ \dfrac{\partial a_k^h}{\partial \mathbf{P}_n} \right]^T. \end{cases} \end{aligned} \tag{12}$$

The gradients are used to express the influence of states on actions for our model, even the gradient is zero, the reference difference allows neurons to transmit important signals. Then, the average marginal contribution value of each feature to the output actions in the datasets can be expressed as

$$\begin{aligned} S(x_i) &= \frac{1}{K} \sum_{k=1}^{K} C_{\Delta x_i^k \Delta a_k^h} \\ &= \frac{1}{K} \sum_{k=1}^{K} \left(\nabla f\left(\mathbf{x}_i^k\right) \times \frac{\partial \mathbf{A}_k}{\partial \mathbf{x}_i^k} \right) \\ &= \frac{1}{K} \sum_{k=1}^{K} \left(\left(\mathbf{x}_i^k - \mathbf{x}_i^{\text{baseline}}\right) \times \frac{\partial \mathbf{A}_k}{\partial \mathbf{x}_i^k} \right) \end{aligned} \tag{13}$$

where $S(x_i)$ is the SHAP value of feature x_i. For each output, both positive and negative contributions need to considered, which can be expressed as

$$\begin{aligned} S(x_i)^+ &= \frac{1}{2}\left(f\left(\mathbf{x}_i^{\text{baseline}} + \Delta \mathbf{x}_i^+\right) - f\left(\mathbf{x}_i^{\text{baseline}}\right) \right) \\ &\quad + \frac{1}{2}\left(f\left(\mathbf{x}_i^{\text{baseline}} + \Delta \mathbf{x}_i^- + \Delta \mathbf{x}_i^+\right) \right. \\ &\qquad \left. - f\left(\mathbf{x}_i^{\text{baseline}} + \Delta \mathbf{x}_i^-\right) \right) \end{aligned} \tag{14}$$

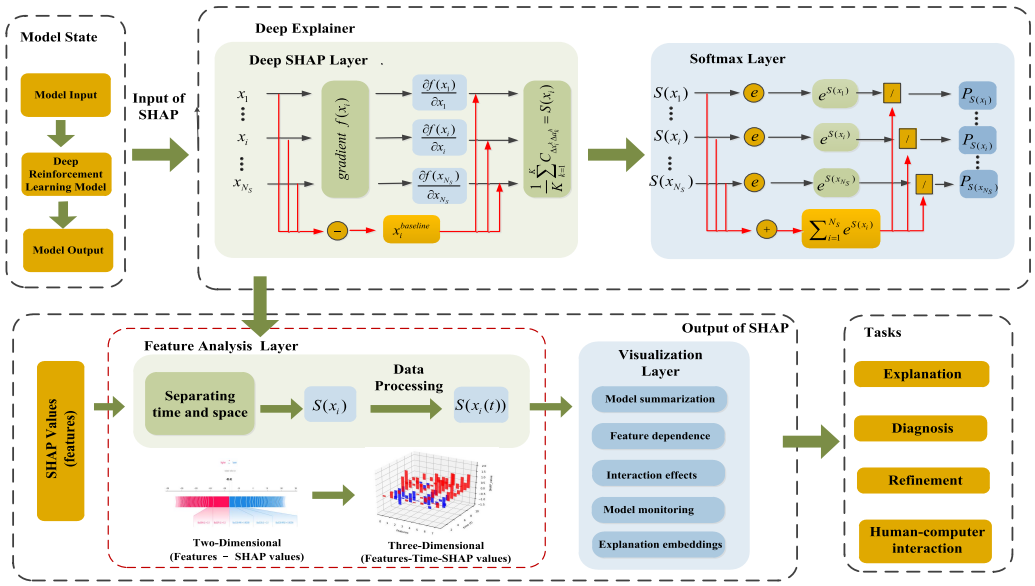


Fig. 2. XAI algorithm framework based on the Deep-SHAP method.

$$S(x_i)^- = \frac{1}{2}\left(f\left(\mathbf{x}_i^{\text{baseline}} + \Delta\mathbf{x}_i^-\right) - f\left(\mathbf{x}_i^{\text{baseline}}\right)\right) + \frac{1}{2}\left(f\left(\mathbf{x}_i^{\text{baseline}} + \Delta\mathbf{x}_i^+ + \Delta\mathbf{x}_i^-\right) - f\left(\mathbf{x}_i^{\text{baseline}} + \Delta\mathbf{x}_i^+\right)\right). \quad (15)$$

The SHAP values of features are the outputs of the XAI model, the contributions are satisfying local accuracy and consistency. These contributions can be applied in the visualization layer, including model summarization, feature dependence, interaction effects, model monitoring, and explanation embedding. Furthermore, to reveal the meaning behind the numbers of SHAP values, the SHAP values are transformed into the form of probability through the softmax function. The softmax function, known as the normalized exponential function, is expressed as

$$P_{S(x_i)} = \text{softmax}(S(\mathbf{X_F})_i) = \frac{e^{S(x_i)}}{\sum_{i=1}^{N_S} e^{S(x_i)}} \quad \text{for } i = 1, \ldots, N_S. \quad (16)$$

Through the softmax function, the output values can be transformed into probability distribution with a range of [0, 1] with the summation of their transformed values equal to 1.

It is worth noting that, the SHAP value method can be adopted to the models with independent input features and sort the contribution of all input features. However, the DRL model under investigation has the characteristics of multidimension and interdependent features, and both temporal and spatial features are interlaced in the input features. It is difficult to explain the decision recommendations by mixed features, and feature analysis is needed, the detail will be presented in Section IV.

IV. Model Interpretation

In this section, the Deep-SHAP method is implemented in Python with the TensorFlow framework. The experimental dataset is generated through the power system simulation and control module, the total number of interaction steps in training is 1 200 000, including 106 800 000 input data $\mathcal{R}_\mathbf{S}$ and 9 600 000 output data $\mathcal{R}_\mathbf{A}$

$$\mathcal{R}_\mathbf{S} = \left\{\mathbf{S}_1 \quad \cdots \quad \mathbf{S}_k \quad \cdots \quad \mathbf{S}_{1,200,000}\right\} \quad (17)$$

$$\mathcal{R}_\mathbf{A} = \left\{\mathbf{A}_1 \quad \cdots \quad \mathbf{A}_k \quad \cdots \quad \mathbf{A}_{1,200,000}\right\}. \quad (18)$$

From the source dataset $\mathcal{R}_\mathbf{S}$ and $\mathcal{R}_\mathbf{A}$, for convenience of calculation and analysis, 1000 cases $\mathcal{R}_{\mathbf{S}_{\text{child}}}$ are selected as the random inputs of the proposed deep explainer

$$\mathcal{R}_{\mathbf{S}_{\text{child}}} = \left\{\mathbf{S}_1 \quad \cdots \quad \mathbf{S}_k \quad \cdots \quad \mathbf{S}_{1,000}\right\}, \quad \mathcal{R}_{\mathbf{S}_{\text{child}}} \in \mathcal{R}_\mathbf{S}. \quad (19)$$

The DRL model for undervoltage load shedding is a multi-classification model, there are 83 input features $\mathbf{X_F}$ and eight types of output actions $\mathbf{A}_k$, which can be expressed as

$$\mathbf{X_F} = \left[x_1 \quad \cdots \quad x_i \quad \cdots \quad x_{83}\right]^T \quad (20)$$

$$\mathbf{A}_k = \left[a_k^1 \quad a_k^2 \quad \cdots \quad a_k^8\right]^T. \quad (21)$$

According to the experimental data, combined with (2)–(5), the input matrix of the deep explainer can be expressed as

$$\mathbf{S}_k = [\,\mathbf{V}_m(t-9) \quad \cdots \quad \mathbf{V}_m(t) \quad \mathbf{P}_n\,]^T \quad (22)$$

$$\mathbf{V}_m(t) = \left[\,V_{4-t} \quad V_{504-t} \quad V_{7-t} \quad V_{507-t} \quad V_{8-t} \quad V_{508-t} \quad V_{18-t} \quad V_{518-t}\,\right]^T \quad (23)$$

$$\mathbf{P}_n = \left[P_4 \quad P_7 \quad P_{18}\right]^T. \quad (24)$$

The meaning of symbols is explained in Table I in detail.

A. Global Explanation

The Deep-SHAP method can provide a global explanation for our DRL model, the influence of all input features $\mathbf{X_F}$ and the results of all output actions $\mathbf{A}_k$ in the model can be computed and presented. To obtain an overview of which

TABLE I
DESCRIPTION OF INPUT FEATURES

Symbol	Meaning
Bus$i$-j	Represents the observed value of voltage magnitude in high-voltage side for the ith bus at jth moment
Bus5i-j	Represents the observed value of voltage magnitude in low-voltage side for the ith bus at jth moment
Pi	Represents the observed value of the percentage load at the ith bus
Action "000"	Means no load reduction
Action "001"	Means 20% load reduction of bus 4
Action "010"	Means 20% load reduction of bus 7
Action "011"	Means 20% load reduction of both buses 4 and 7
Action "100"	Means 20% load reduction of bus 18
Action "101"	Means 20% load reduction of both buses 4 and 18
Action "111"	Means 20% load reduction of both buses 4, 7 and 18

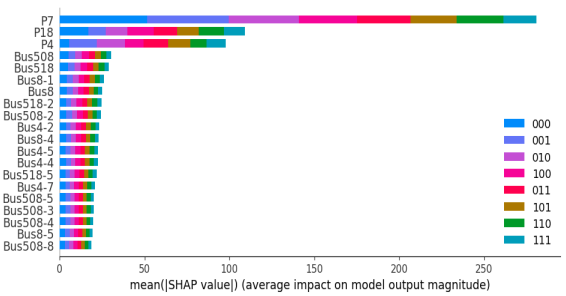


Fig. 3. Rich summaries of entire model based on SHAP.

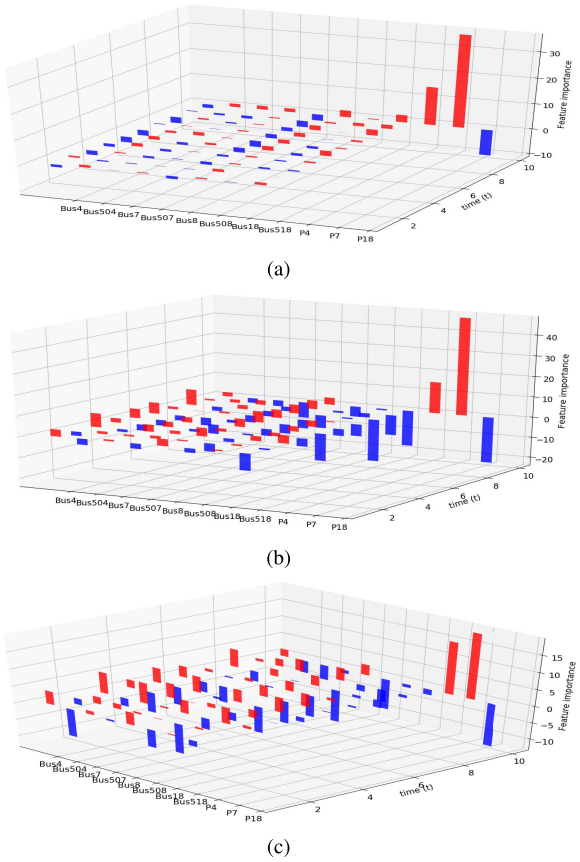


Fig. 4. (a) Influence of features in individual example for the case of no load reduction. (b) Influence of features in individual example for the case of 20% load reduction of bus 4. (c) Influence of features in individual example for the case of 20% load reduction of bus 7.

features are most important for the model, one can plot the $S(x_i)$ of random feature x_i for the input matrix $\mathbf{S}_k$, $\mathbf{S}_k \in \mathcal{R}_\mathbf{S}$. Fig. 3 is the summary plot, which take the mean absolute value of the $S(x_i)$ for each feature x_i to get a stacked bars for multiclass outputs $\mathbf{A}_k$, and uses $S(x_i)$ to show the distribution of feature x_i's impacts on the model outputs. In Fig. 3, the horizontal axis is the average impact on output magnitude, the vertical axis represents the most influential 20 features of $\mathbf{X_F}$. The eight colors in Fig. 3 represent eight output actions, respectively. The meaning of symbols in Fig. 3 is explained in Table I in detail.

In Fig. 3, each input feature x_i is sorted after computing, the features of great contribution under multiclassification output can be seen intuitively and globally, under these circumstances, operators can obtain important information for decision-making. However, the input features $\mathbf{X_F}$ are independent and include temporal and spatial properties. It is not intuitive to consider all features in a 2-D graph as the interaction between features and the results of the XAI model will be difficult to understand. Therefore, features need to be categorized to distinguish their temporal and spatial properties.

B. Local Explanation

Among the 1000 randomly selected cases $\mathcal{R}_{\mathbf{S}_{\text{child}}}$, the training outputs $\mathbf{A}_k$ of the model only have three categories, they are no load reduction, 20% load reduction of bus 4 and 20% load reduction of bus 7, respectively. The temporal and spatial properties of input features $\mathbf{X_F}$ are demonstrated using the 3-D graph containing time t, feature x_i and feature importance $S(x_i)$ as shown in Fig. 4. Fig. 4(a) is a case with the output of no load reduction, Fig. 4(b) is a case with the output of 20% load reduction of bus 4, and Fig. 4(c) is a case with the output of 20% load reduction of bus 7. The contribution of x_i at each moment t can be seen clearly, x_i increasing the prediction are shown in red, and those decreasing the prediction are shown in blue, any time t can be selected for feature analysis, we choose x_i of the current moment ($t = 10$) for convenience. Combined with (22)–(24), the input matrix can be expressed as

$$\mathbf{S}_k = \begin{bmatrix} V_4 & V_{504} & V_7 & V_{507} & V_8 \\ & & V_{508} & V_{18} & V_{518} & P_4 & P_7 & P_{18} \end{bmatrix}^T. \quad (25)$$

Considering the multiclassification outputs $\mathbf{A}_k$, cases from each of the three output types in our trained DRL model are selected to show the visualization of interpretation, which are presented in Fig. 5. Fig. 5(a), (c), and (e) are the bar charts of the average $S(x_i)$. Fig. 5(b), (d) and (f) are a set of bee swarm plots, each dot corresponds to an individual data in the study, the color represents the feature value, red color means positive impacts, i.e., increasing the prediction, while blue means negative impact. When the recommendation is no load sheddin, as shown in Fig. 5(a) and (b), the load conditions of buses 7, 18, and 4 are the three most influential features contributing to this recommendation. Moreover, the contribution will decrease if the value of the load on bus

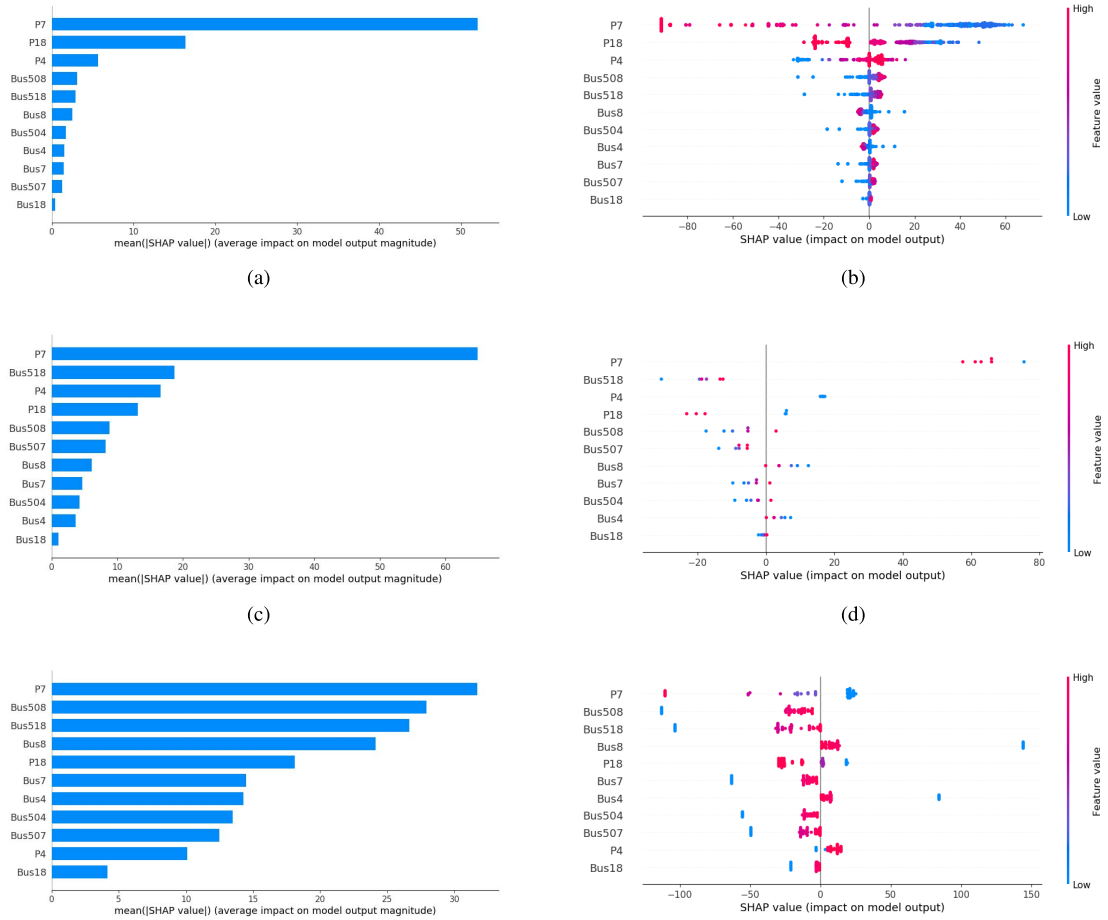


Fig. 5. (a) Bar chart of the average SHAP value magnitude for no load reduction. (b) Bee swarm plot of SHAP value magnitude for no load reduction. (c) Bar chart of the average SHAP value magnitude for 20% load reduction of bus 4. (d) Bee swarm plot of SHAP value magnitude for 20% load reduction of bus 4. (e) Bar chart of the average SHAP value magnitude for 20% load reduction of bus 7. (f) Bee swarm plot of SHAP value magnitude for 20% load reduction of bus 7.

7 increases. Similarly, in Fig. 5(c) and (d), the three most influential features are the load condition of bus 7, the voltage magnitude of bus 18 at the low voltage side, and the load condition of bus 4. When the recommendation output is to reduce a load of bus 7 by 20%, as shown in Fig. 5(e) and (f), the load condition of bus 7, the voltage magnitude of buses 8 and 18 at the low voltage side is more important than other features. In all the three different output categories, the most influential feature is the load of bus 7.

For features selected in the $\mathbf{S}_k$, $\mathbf{S}_k \in \mathcal{R}_\mathbf{S}$, the Deep-SHAP method can be used to provide the local interpretability. For a specific input matrix $\mathbf{S}_k$, the reasons why the model makes certain decisions can be given by local interpretability. Fig. 6 shows how much each feature x_i is pushing the output $\mathbf{A}_k$ of the DRL model from the base value, which is the average value for each type of output a_k^h. The six most influential feature values in Fig. 6 and their importance $S(x_i)$ are listed in Table II.

Fig. 6(a) is a case with the recommendation output of no load reduction. In this scenario, the voltage magnitudes are over 0.95 p.u., which is consistent with the common decision of a human operator. Fig. 6(a) and (b) are the

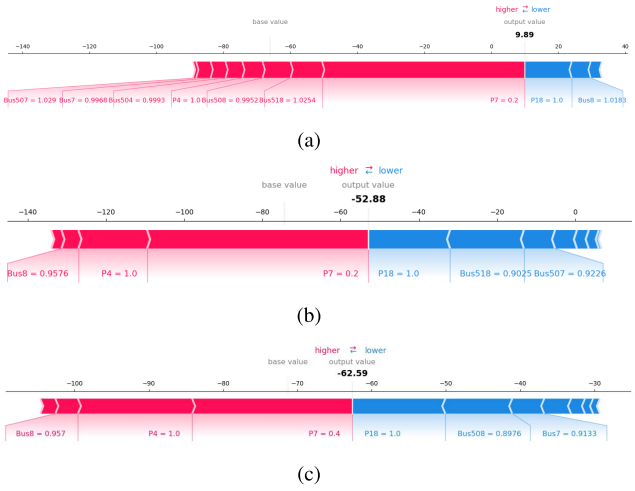


Fig. 6. (a) Influence of features for the case of no load reduction. (b) Influence of features for the case of 20% load reduction of bus 4. (c) Influence of features for the case of 20% load reduction of bus 7.

cases recommending 20% load reduction of buses 4 and 7 respectively, the loads of buses 4 and 7 being the two

TABLE II
SIX MOST INFLUENTIAL FEATURES IN FIG. 6 AND THEIR IMPORTANCE

$\mathbf{S}_k$	x_i	x_i' value	$S(x_i)$	$P_{S(x_i)}$
Case 1	P7	0.4	22.911	31.0%
	P4	1.0	14.511	19.7%
	P18	1.0	-12.511	-16.9%
	Bus508	0.898	-8.72	-11.8%
	Bus8	0.957	2.722	3.7%
	Bus518	0.991	-1.359	-1.8%
Case 2	P7	0.2	61.183	43.0%
	P18	1.0	-20.375	-13.2%
	Bus518	0.903	-18.783	-14.3%
	P4	1.0	16.648	11.7%
	Bus508	0.938	-5.282	-3.7%
	Bus8	0.958	3.845	2.7%
Case 3	P7	0.2	52.353	58.4%
	P18	1.0	-9.443	-10.5%
	P4	1.0	5.434	6.1%
	Bus508	0.995	4.386	4.9%
	Bus518	1.025	4.278	4.8%
	Bus8	1.018	-3.951	-4.4%

most influential features. The output value in Fig. 6 is the cumulative return value for output action a_k^h. The features in favor of the recommendation output are shown in red, and those not in favor of the recommendation output are shown in blue. Moreover, some voltage magnitudes that do not return to the standard value within the specified time are presented, which means load shedding is needed for stable operation. Such pieces of evidence make operators in the power system more convinced with the judgments of our DRL model.

However, SHAP value $S(x_i)$ is not intuitive to show the interpretability for human operators. In this way, we transform $S(x_i)$ into the form of probability $P_{S(x_i)}$ through softmax function, which is much easier to understand by human operators as in Table II. Through $P_{S(x_i)}$, the influence of features on the results can be interpreted much more easily.

The above analysis can be used as an important basis for the decision-making of operators in the power system. The interpretability of the proposed DRL model is an implicit solution to understand human–computer interaction. In the process of real-world power system dispatch, although the DRL model can help provide actionable recommendations in emergency control events, the final decision must be made by operators per the requirement of policy and regulations. The interpretation of XAI provides additional information and explanation for the operators to make better decisions and deployments empowered by AI.

V. CONCLUSION

AI has become an important means of decision-making for power systems, as a result, providing informative and reliable XAI technologies for power system management and control also becomes urgent and necessary. Specifically, the interpretability issue in the DRL model for power system control has been discussed in this article. The Deep-SHAP method is proposed and adopted to provide a reasonable interpretable model by calculating the importance of input features based on a backpropagating strategy. Through the classification and probabilistic feature analysis, it aims to provide sufficient decision-making assistance for human operators. The cognitive ability of the XAI model is better than that of the operator, which helps to find problems in data and model training, and improves the performance of the ML model. The application of the Deep-SHAP method in the DRL model of power system emergency control is implemented and analyzed.

The research on XAI technologies in power systems is still at its beginning stage. There are still many problems that need to be considered to truly help the operators to understand the decision-making process, such as how to provide an explanation related to the power flow and dynamic process of the power system in the proposed DRL model in this article. Future research will be conducted to combine symbolism and connectionism and build a heterogeneous information network (HIN) for power systems. The graph network structure has more abundant semantic knowledge and more intuitive relationship information, by combining HIN in power systems and XAI models, better interpretation can be obtained for human operators.

REFERENCES

[1] J. Zhang, C. Lu, J. Si, J. Song, and Y. Su, "Deep reinforcement leaming for short-term voltage control by dynamic load shedding in China southem power grid," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2018, pp. 1–8.

[2] Z. Yan and Y. Xu, "A multi-agent deep reinforcement learning method for cooperative load frequency control of a multi-area power system," *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4599–4608, Jun. 2020.

[3] W. Liu, D. X. Zhang, X. Y. Wang, J. Hou, and L. P. Liu, "A decision making strategy for generating unit tripping under emergency circumstances based on deep reinforcement learning," *Proc. CSEE*, vol. 38, no. 1, pp. 109–119, 2018.

[4] D. T. V. Franc, D. E. ois Lavet, and R. Fonteneau, "Deep reinforcement learning solutions for energy microgrids management," in *Proc. Eur. Workshop Reinforcement Learn.*, 2016, pp. 1–7.

[5] P. D. Xu, Y. Z. Pei, X. H. Zheng, and J. Zhang, "A simulation-constraint graph reinforcement learning method for line flow control," in *Proc. 4th IEEE Conf. Energy Internet Energy Syst. Integr.*, Nov. 2020, pp. 319–324.

[6] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138–52160, 2018.

[7] Q.-S. Zhang and S.-C. Zhu, "Visual interpretability for deep learning: A survey," *Frontiers Inf. Technol. Electron. Eng.*, vol. 19, no. 1, pp. 27–39, 2018.

[8] M. Ribeiro, S. Singh, and C. Guestrin, "'Why should i trust you?' Explaining the predictions of any classifier," in *Proc. 22nd Acm Sigkdd Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 1135–1144.

[9] M. Hind *et al.*, "TED: Teaching AI to explain its decisions," in *Proc. AAAI/ACM Conf. AI, Ethics, Soc.*, Jan. 2019, pp. 123–129.

[10] M. Wu *et al.*, "Beyond sparsity: Tree regularization of deep models for interpretability," 2017, *arXiv:1711.06178*. [Online]. Available: https://arxiv.org/abs/1711.06178

[11] O. Bastani, C. Kim, and H. Bastani, "Interpreting blackbox models via model extraction," in *Proc. Conf. Comput. Res. Repository (CoRR)*, 2017, pp. 1–28. [Online]. Available: http://arxiv.org/abs/1705.08504

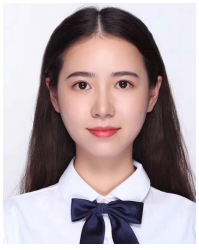
[12] S. H. Luo, X. C. Wang, G. F. Fang, Y. Hu, D. P. Tao, and M. L. Song, "Knowledge amalgamation from heterogeneous networks by common feature learning," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, 2019, pp. 3087–3093.

[13] Z. Che, S. Purushotham, R. Khemani, and Y. Liu, "Distilling knowledge from deep networks with applications to healthcare domain," *Ann. Chirurgie*, vol. 40, no. 8, pp. 529–532, 2015.

[14] J. Song, Y. X. Chen, X. C. Wang, C. C. Shen, and M. L. Song, "Deep model transferability from attribution maps," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 6182–6192.

[15] M. Alshammari, O. Nasraoui, and S. Sanders, "Mining semantic knowledge graphs to add explainability to black box recommender systems," *IEEE Access*, vol. 7, pp. 110563–110579, 2019.

[16] K. Yang, X. Kong, Y. Wang, J. Zhang, and G. De Melo, "Reinforcement learning over knowledge graphs for explainable dialogue intent mining," *IEEE Access*, vol. 8, pp. 85348–85358, 2020.
[17] D. L. Marino, C. S. Wickramasinghe, and M. Manic, "An adversarial approach for explainable AI in intrusion detection systems," in *Proc. 44th Annu. Conf. Ind. Electron. Soc.*, 2018, pp. 3237–3243.
[18] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart, "Retain: An interpretable predictive model for healthcare using reverse time attention mechanism," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 3504–3512.
[19] G. R. Vásquez-Morales, S. M. Martínez-Monterrubio, P. Moreno-Ger, and A. J. Recio-García, "Explainable prediction of chronic renal disease in the colombian population using neural networks and case-based reasoning," *IEEE Access*, vol. 7, pp. 152900–152910, 2019.
[20] C. Molnar. (2018). *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*. [Online]. Available: https://christophm.github.io/interpretable-ml-book/
[21] H. Joo and K. Kim, "Visualization of deep reinforcement learning using grad-CAM: how AI plays atari games?" in *Proc. IEEE Conf. Games (CoG)*, Oct. 2019, pp. 1–2.
[22] T. Spinner, U. Schlegel, H. Schäfer, and M. El-Assady, "Explainer: A visual analytics framework for interactive and explainable machine learning," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 1, pp. 1064–1074, Jan. 2020.
[23] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," 2017, *arXiv:1704.02685*. [Online]. Available: http://arxiv.org/abs/1704.02685
[24] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 30, 2017, pp. 4765–4774.
[25] M. R. Karim, M. Cochez, O. Beyan, S. Decker, and C. Lange, "OncoNetExplainer: Explainable predictions of cancer types based on gene expression data," in *Proc. IEEE 19th Int. Conf. Bioinf. Bioeng. (BIBE)*, Oct. 2019, pp. 415–422.
[26] S. M. Lundberg *et al.*, "From local explanations to global understanding with explainable AI for trees," *Nature Mach. Intell.*, vol. 2, no. 1, pp. 56–67, 2020.
[27] A. Messalas, Y. Kanellopoulos, and C. Makris, "Model-agnostic interpretability with shapley values," in *Proc. 10th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Jul. 2019, pp. 1–7.
[28] M. Kuzlu, U. Cali, V. Sharma, and O. Guler, "Gaining insight into solar photovoltaic power generation forecasting utilizing explainable artificial intelligence tools," *IEEE Access*, vol. 8, pp. 187814–187823, 2020.
[29] Q. Huang, R. Huang, W. Hao, J. Tan, R. Fan, and Z. Huang, "Adaptive power system emergency control using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1171–1182, Mar. 2020.
[30] K. Zhang, P. D. Xu, and J. Zhang, "Explainable AI in deep reinforcement learning models: A shap method applied in power system emergency control," in *Proc. 4th IEEE Conf. Energy Internet Energy Syst. Integr.*, 2020, pp. 711–716.
[31] V. Chuvychin, N. Gurov, and S. Kiene, "Application of new emergency control principle in power systems," in *Proc. IEEE Bucharest PowerTech*, Jun. 2009, pp. 1–6.
[32] H. Hagras, "Toward human-understandable, explainable AI," *Computer*, vol. 51, no. 9, pp. 28–36, Sep. 2018.

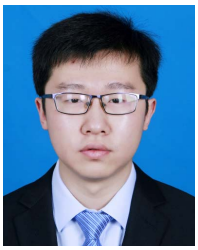


Ke Zhang (Student Member, IEEE) received the M.S. degree in power electronics and electric drives from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2018. She is currently pursuing the Ph.D. degree in electrical engineering with Wuhan University, Wuhan, China.

Her research interest covers artificial intelligence, smart grids, and trustworthy artificial intelligence (AI).

Jun Zhang (Senior Member, IEEE) received the bachelor's and master's degrees in electrical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2003 and 2005, respectively, and the Ph.D. degree in electrical engineering from Arizona State University, Tempe, AZ, USA, in 2008.

He is currently a Professor at the School of Electrical Engineering and Automation, Wuhan University, Wuhan. His research interest covers intelligent systems, artificial intelligence, knowledge automation, and their applications in intelligent power and energy systems.



Pei-Dong Xu (Graduate Student Member, IEEE) received the master's degree from the School of Electrical Engineering, Wuhan University, Wuhan, China, in 2018, where he is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Automation.

His research interests cover artificial intelligence and smart grids.

Tianlu Gao (Member, IEEE) received the master's degree from the School of Electrical and Computer, University of Denver, Denver, CO, USA, in 2019.

He is currently a Research Engineer at the School of Electrical and Automation, Wuhan University, Wuhan, China. His main research interests are the application of AI, NLP, and blockchain in power system operation and control.

David Wenzhong Gao (Fellow, IEEE) received the M.S. and Ph.D. degrees in electrical and computer engineering, specializing in electric power engineering, from the Georgia Institute of Technology, Atlanta, GA, USA, in 1999 and 2002, respectively.

He is now with the Department of Electrical and Computer Engineering, University of Denver, Denver, CO, USA. His current teaching and research interests include renewable energy and distributed generation, microgrids, smart grids, power system protection, power electronics applications in power systems, power system modeling and simulation, and hybrid electric propulsion systems.

Dr. Gao was the General Chair of the 48th North American Power Symposium (NAPS 2016) and the IEEE Symposium on Power Electronics and Machines in Wind Applications (PEMWA 2012). He is an Associate Editor of the IEEE JOURNAL OF EMERGING AND SELECTED TOPICS IN POWER ELECTRONICS and the *Journal of Modern Power System and Clean Energy*. He was an Editor of the IEEE TRANSACTIONS ON SUSTAINABLE ENERGY.