*Sensors and Materials*, Vol. 35, No. 8 (2023) 3057–3069
MYU Tokyo

S & M 3374

# Load Identification of Low-voltage Station Area Based on Power Metering Sensor and Deep Reinforcement Learning

Qiang Fu,[1] Xiaohua Yang,[2] Jianyu Ren,[2] Yonghui Zhao,[2] Hao Yang,[2] and Shibo Pan[3*]

[1]Lijiang Power Supply Bureau of Yunnan Power Grid Co., Ltd., Lijiang 674100, China
[2]Measurement Center, Yunnan Power Grid Co., Ltd., Kunming 650051, China
[3]Faculty of Civil Aviation and Aeronautics, Kunming University of Science and Technology, Kunming 650500, China

(Received April 18, 2023; accepted August 10, 2023)

***Keywords:*** smart grid, load identification in station area, load clustering, density peak clustering, end-to-side neural network

With the continuous development of smart grid technology, people's demand for smart electricity consumption is increasing, and electricity consumption identification is a key aspect of achieving smart electricity consumption. Therefore, to promote the development of electricity consumption identification, we have studied load identification methods in low-voltage (LV) station areas and proposed a set of load identification models based on deep reinforcement learning. Each model consists of a non-intrusive load monitoring (NILM) device, an improved adaptive density peak (ISDPC) model, a new end-to-side neural network called GhostNet, and a data processing and analysis module specific to 10 kV power transmission. Considering the complexity and diversity of loads in the 10 kV power transmission system, we employ ISDPC algorithm to perform cluster analysis on load characteristic data and use GhostNet for load identification. Additionally, we preprocess and extract features from the data specific to 10 kV power transmission to improve the accuracy and effectiveness of the identification. Finally, we compare our results with those of the *k*-means clustering algorithm, Euclidean distance load curve clustering, and other algorithms to demonstrate the superiority of our method in terms of clustering and identification accuracies.

## 1. Introduction

In recent years, with the continuous development of smart grid technology, there has been an increasing demand for smart electricity consumption. Electricity consumption identification technology is a crucial aspect of achieving intelligent electricity consumption. Accurately monitoring and identifying 10 kV electricity consumption equipment can assist power suppliers in allocating electricity resources more efficiently. Furthermore, for the 10 kV power transmission system, the accurate collection and identification of load information, as well as the timely response to load demands, are highly important. By accurately identifying the loads in the 10 kV power transmission system, the real-time monitoring and management of electricity

*Corresponding author: e-mail: 1027949014@qq.com
https://doi.org/10.18494/SAM4517

ISSN 0914-4935 © MYU K.K.
https://myukk.org/

consumption can be achieved.(1) This facilitates a better understanding of consumption patterns and behaviors of 10 kV loads for smart grid system managers, providing valuable insights for electricity supply–demand balance, energy scheduling, and energy efficiency optimization.(2)

As the key to load identification, clustering analysis has been of great interest to researchers as it plays an important role in data mining and analysis.(3) The clustering operation can divide a large amount of discrete data into different clusters by calculating the similarity between the data in accordance with those distribution.(4) On the basis of different algorithmic principles, the current common clustering algorithms can be broadly classified into division-based clustering, hierarchy-based clustering, density-based clustering, grid-based clustering, and model-based clustering.(4,5) Hierarchy-based clustering creates a hierarchical nested clustering tree by calculating the similarity between data points of different categories. By dividing the data space into grid cells, the data objects are mapped to grid cells, and the density of each cell is calculated. Density-based clustering algorithms can find clusters of various shapes and sizes in different data, and the idea is that after selecting high-density points, the surrounding points that are similar to the high-density points are clustered into one class.(6)

The density peak clustering (DPC) algorithm is a simple and effective clustering algorithm that maps arbitrary-dimensional data to two dimensions and constructs hierarchical relationships between data in the reduced dimensional space, from which it is very easy to select high-density data points far away from other high-density areas. These points are called peak density points and can be used as clustering centers. On the basis of the constructed hierarchical relationship, the algorithm provides two different methods to complete the final clustering.(7,8) This algorithm can effectively cluster complex data and has a good clustering effect on any data set. At present, DPC is widely used in various fields, such as machine learning and image processing.(9)

Although the DPC algorithm has the above-described advantages, it also has some disadvantages; for example, the truncation distance cannot be automatically adjusted and the cluster center needs to be manually specified. The adaptive DPC (ADPC) algorithm with an adaptive acquisition mechanism can well solve these disadvantages.(10) The ADPC algorithm can automatically adjust the truncation distance in accordance with the problem characteristics. The automatic acquisition of clustering centers greatly reduces the effect of human error. Liu and Xu(11) proposed a fuzzy *C*-means clustering algorithm optimized by the density peak algorithm, which can adaptively generate the initial cluster center, determine the number of clusters, and optimize the convergence process of the algorithm, which has certain guiding significance. For a truncation distance that cannot be adjusted automatically, Wang *et al.*(12) proposed a new adaptive aggregation strategy, whereby the initial cluster center is first determined by giving a threshold, then the remaining points are allocated in accordance with the nearest distance to the initial cluster center, and finally, similar clusters are merged in accordance with the achievable density between clusters. Qian and Jin(13) proposed a new adaptive aggregation strategy, in which the initial class cluster center is first determined by giving a threshold value to the algorithm, then the remaining points are assigned on basis of the nearness to the initial class cluster center, and finally, similar class clusters are merged in accordance with the density between class clusters up to the adaptively determined truncation distance parameter by minimizing the information entropy.

In addition, intelligent optimization algorithms are also useful for selecting the appropriate truncation distance and improving the accuracy of clustering. As a new intelligent optimization algorithm, the whale optimization algorithm (WOA) has many advantages compared with other optimization algorithms.[14] Liu *et al.*[15] proposed a new three-stage hybrid feature gene selection method comprising a variance filter, a polar random tree, and the WOA. The optimal feature gene subset is selected by the WOA. Experimental results show that this method has significant advantages in a variety of evaluation indexes. By combining adaptive weighting and the WOA, improvements were made using Cauchy mutation[16] and simulated annealing strategies.[17] These improvements were aimed at addressing the issue of the WOA's susceptibility to local optima, and the algorithm's strong combinational capability after incorporating these modifications was demonstrated. Zhang *et al.*[18] proposed an efficient intelligent prediction model based on the machine learning method. This model comprises an improved WOA (RRWOA) based on random contraction strategy (RCS) and the Rosenbrock method combined with the K-nearest-neighbor (KNN) classifier. The problem of the WOA falling into a local optimum was solved, and the experimental results showed that the improved RRWOA achieves good results.

After completing the clustering operation, the clustering centers that have been classified must be identified to match the different clustering centers to the user appliances; the neural network GhostNet would be a solution to this problem. As a new neural network model, the computational effort required by GhostNet is greatly reduced compared with traditional neural networks, and the complexity of the convolution operation is largely optimized.[19,20] Furthermore, the use of Ghost modules instead of the convolutional layers in the convolutional neural network structure in other studies[21] has improved the objective image quality evaluation, proving the excellence of the GhostNet performance. Gao and Wang[22] designed a lightweight network, Ghostnet, as the backbone network of the Deblurganv2 generator. It has a reduced number of network parameters and a ruduced amount of calculation, and it effectively improved the wind power detection image processing efficiency. Zheng *et al.*[23] proposed the Little-YOLOv4 (you only look once version 4) network structure in which GhostNet was used to extract image features, and BiFPN path fusion was added to improve the path aggregation network (PANet) to integrate richer semantic features and retain spatial information.

In summary, we propose a load identification method based on deep reinforcement learning for a low-voltage (LV) station area. First, the load characteristics data of LV substation users are obtained using nondestructive testing equipment, and then the load characteristics are clustered by an improved adaptive density peak model combined with the WOA to form load datasets with different characteristics, after which the datasets are input into GhostNet for load identification. Then, we conduct a cross-sectional comparison of our method with the *k*-means clustering algorithm and Euclidean distance load curve clustering. The superiority of our method in terms of clustering and identification accuracies is proved.

## 2. Improved ADPC Algorithm Combined with WOA

The clustering algorithm in our study is improved in the following two aspects: firstly, the automatic selection of clustering centers is achieved on the basis of the trend of the slope change of the weighted local density and relative distance product; secondly, the truncation distance $d_c$ in the DPC algorithm is optimized by using the stronger merit-seeking ability of the WOA.

### 2.1 Principle of DPC algorithm

DPC is mainly based on two intuitive assumptions: (1) the cluster center is surrounded by a group of neighboring points with a lower density; (2) the distance of the cluster center from the points with a density higher than it is relatively large. The algorithm requires only one parameter, the truncation distance $d_c$, to manually select the clustering center points, and then completes the clustering and assignment of the remaining points by a one-step assignment strategy. In the DPC algorithm, the local density is calculated in two ways, and when the dataset is large, the local density is calculated as

$$\rho_i = \sum_{j \neq i} x(d_{ij} - d_c), \tag{1}$$

where $\begin{cases} x(x)=1, x<0 \\ x(x)=0, x\geq 0 \end{cases}$ and $d_{ij}$ is the Euclidean distance to the other points, defined as

$$d_{ij} = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})}. \tag{2}$$

When the dataset is small, the local density is defined as

$$\rho_i = \sum_{j \neq i} \exp\left( -\frac{d_{ij}^{\,2}}{d_c^{\,2}} \right), \tag{3}$$

where the truncation distance is $d_c = d_i \times 2\%$ and $d_{ij}$ is the ascending order of the Euclidean distance between any two points among all data points.

Define the relative distance as the distance between the sample point and the nearest high-density sample point:

$$\delta_i = \begin{cases} \max\limits_{j:\rho_i > \rho_j} di_j, \\ \max\limits_{j:\rho_i < \rho_j} di_j. \end{cases} \tag{4}$$

The DPC algorithm, based on the derived truncation distance $d_c$, then substitutes it into the formula of the local density $\rho$ and the distance $\delta$, and draws a decision diagram in two-dimensional space based on $\rho$ and $\delta$. The larger values of $\rho$ and $\delta$ are manually selected as clustering centers using the decision diagram, and finally, by a one-step allocation strategy, the remaining points are assigned to the nearest class clusters with a higher density.

### 2.2 Adaptive clustering center selection

The original DPC algorithm needs to manually intercept the points with the local density $\rho$ and the large relative distance $\delta$ as clustering centers in the decision diagram when selecting the clustering centers, and the interception process is subjective. When facing the user load data of a LV station area, manually setting the clustering centers (i.e., appliance types) is usually not comprehensive enough, and if there is an error in selecting the clustering centers, it will cause the whole clustering step to fail to achieve the desired results, so we need to solve this problem by automatically selecting the clustering centers through the adaptive selection strategy.

In the adaptive selection strategy, considering the local density $\rho$ and the relative distance $\delta$ together, we define

$$\gamma_i = \rho_i \delta_i ,\ i = 1, 2, N, \tag{5}$$

where $N$ is the number of sample points in the dataset.

Avoiding the interaction between different magnitudes, the normalization of $\rho$ and $\delta$ yields the following new definition of $\gamma$:

$$\gamma_i = \rho_i' \cdot \delta_i' = \frac{(\rho_i - \rho_{\min})(\delta_i - \delta_{\min})}{(\rho_{\max} - \rho_{\min})(\delta_{\max} - \delta_{\min})}. \tag{6}$$

The clustering centers are selected in accordance with the $\gamma$ descending ranking chart, and the points with larger $\gamma$ values are more likely to be clustering centers. This can effectively eliminate the errors caused by considering $\rho$ and $\delta$ separately.

### 2.3 WOA

The WOA is a new bionic algorithm based on simulating whale predation behavior. It is mainly divided into three stages: prey siege stage, bubble attack stage, and prey search stage.

The specific steps of the WOA are as follows.

Step 1: S individual whale positions are randomly initialized in the solution space, and the maximum number of iterations, $T$, and the current number of iterations, $t$, are given.

Step 2: By calculating each fitness value using the objective function and retaining the optimal individual as $X^*(t)$, the mathematical model for updating the whale position to the optimal position is

$$X(t+1) = X^*(t) - A \cdot \left| C \cdot X^*(t) - X(t) \right|, \tag{7}$$

$$A = 2a \cdot r - a, \tag{8}$$

$$C = 2 \cdot r, \tag{9}$$

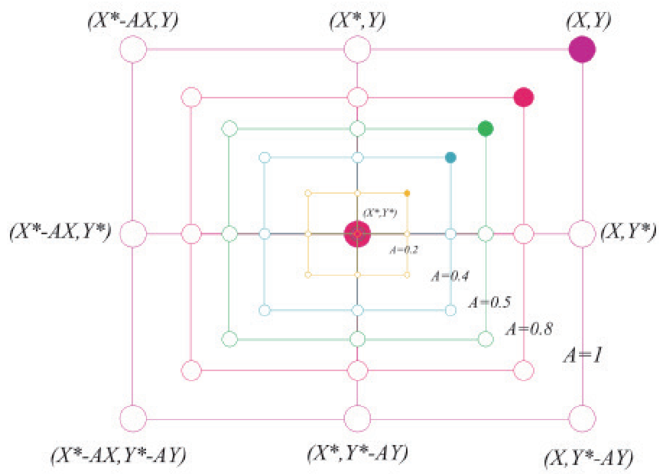



Fig. 1. (Color online) Whale location update.

where $t$ is the number of current iterations, $X^*(t)$ is the current optimal position, $X(t)$ is the current position, $A$ and $C$ are the system vectors, $r$ is a random number between [0, 1], and $a$ is a linearly decreasing parameter from 2 to 0.

Step 3: The coefficient vector A and the random number $p$ are updated, and the corresponding position updates are performed for all individuals in accordance with the values of $A$ and $p$. It is assumed that the probabilities of humpback whale contraction surround and spiral position update are both 50%. The following position update model is established:

$$X(t+1)=\begin{cases} X^*(t)-A\cdot\left|C\cdot X^*(t)-X(t)\right|, & p<0.5 \\ X^*(t)+D'\cdot e^{bl}\cdot\cos(2\pi l), & p\geq 0.5 \end{cases} \tag{10}$$

$$D'=\left|X^*(t)-X(t)\right|, \tag{11}$$

where $b$ is a constant that defines the shape of the logarithmic helix, $l$ is a random number between [−1, 1], and the variable $p$ is a random number between [0, 1].

If $|A| \geq 1$, $p < 0.5$, perform the update of

$$X(t+1)=X_{rand}(t)-A\cdot\left|C\cdot X_{rand}(t)-X(t)\right|, \tag{12}$$

where $X_{rand}$ is a random selection of whale individuals from the current population. During the optimization of the specific problem, the whale individuals keep approaching the optimal solution by different location update methods.

If $|A| < 1$, $p < 0.5$, perform the update of Eq. (7).

If $p \geq 0.5$, update is carried out according to Eq. (13), and the update mechanism diagram is shown in Fig. 1.

$$X\left(t+1\right)=X^{*}\left(t\right)+D'\cdot e^{bl}\cdot\cos\left(2\pi l\right) \tag{13}$$

Step 4: The fitness values of the updated individuals are calculated, and the optimal individual positions and fitness values are updated to retain the optimal individuals.
Step 5: Determine whether $t < T$ holds. If yes, set $t = t + 1$ and return to Step 2; otherwise, the algorithm ends and solution and the value are output.

This algorithm selects the cluster centers on the basis of slope of $\gamma$. The point $\gamma_{i_0}=\arg\max\limits_{i}\left(\left(\gamma_i-\gamma_{i+1}\right)/\left(\gamma_{\max}-\gamma_{\min}\right)\right)$ with the largest slope of [0, 1] is generally selected as the dividing point between the clustering center and other points, but the jump of $(\gamma_i - \gamma_{i+1})/(\gamma_{\max} - \gamma_{\min})$ is too strong to accurately select the clustering center; hence, the slope must be weighted as

$$k_i=(i-1)\frac{\gamma_i-\gamma_{i+1}}{\gamma_{\max}-\gamma_{\min}},i=2,3,\ldots,N. \tag{14}$$

The cut-off distance $d_c$ in the DPC algorithm must be set artificially, and its value is very important. To adapt to the demand of user load identification in the LV station area, we use the WOA with a strong merit-seeking ability to select the best cut-off distance and introduce accuracy (*ACC*) as the objective function of the WOA algorithm. Let $P_j$ be the known manually labeled clusters and $C_j$ be the clusters after clustering; then, *ACC* is calculated as follows:

$$ACC(P_j,C_i)=\frac{\left|P_j\cap C_i\right|}{\left|C_i\right|}. \tag{15}$$

The objective function *ACC* index is a one-dimensional function of $d_c$, that is, given $d_c$, an *ACC* index value can be obtained. The range of the *ACC* index is between [0, 1], and the closer the value is to 1, the better the clustering result is.

The WOA is used to find $d_c$ with the maximum *ACC* index in the DPC algorithm as the optimal $d_c$ for the current dataset, thus realizing the clustering of the algorithm.

## 3. GhostNet Model Construction

After the above clustering operation of the improved ADPC algorithm combined with the WOA, the user load data forms several clustering centers with different load characteristics. These clustering centers correspond to different types of electrical appliances. Now we need to identify these clustering centers through neural networks, so that we finally know which appliances correspond to which clustering centers. In this paper, we use the new neural network GhostNet, which can produce more feature maps through simple operations. Its operation principle is to perform a series of linear transformations through a set of original feature maps, and the ‘ghost feature maps’ of the required information can be discovered from the original features through simple operations. GhostNet can achieve the construction of a Ghost bottleneck by stacking Ghost modules. The dataset processed by clustering is segmented and the points are used to form the corresponding power–time curve. GhostNet is used to identify such a curve through the trained model, so as to know the user 's electricity consumption habits in different time periods. Finally, a visual feature map is output, and the user 's load situation is analyzed by the corresponding evaluation criteria.

### 3.1 Defining the Ghost module

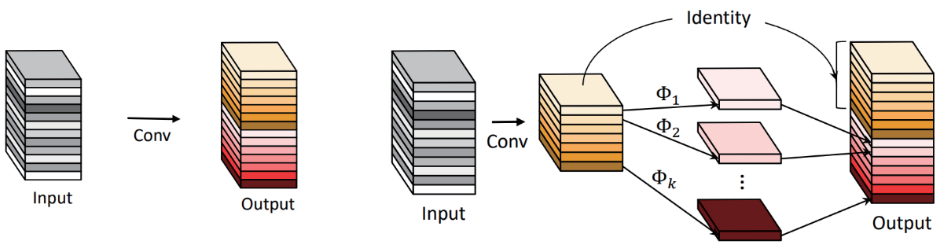
For any $n$ original feature map, $O \in \mathbb{R}^{h \times w \times n}$ is generated using a single convolution calculation.

$$Y = O * f \tag{16}$$

$f \in \mathbb{R}^{c \times k \times k \times n}$ is the convolution kernel of this layer, $k \times k$ is the kernel size of $f$, $Y \in \mathbb{R}^{h^{\cdot} \times w^{\cdot} \times n}$ is the feature output map of the n channels, and $h^{\cdot}$ is the height and $w^{\cdot}$ is the width of the output data. The bias term is omitted here for the simplicity of calculation, and the hyperparameters of this model are the same as those in ordinary convolution in order to ensure a consistent spatial size of the feature maps.

### 3.2 Build GhostNet

First, a Ghost bottleneck is built with a Ghost module, which generally consists of two Ghost modules, one of which is used as an extension layer with an increased number of channels, while the other reduces the number of channels to match the shortcut path. The starting convolution of the module is point convolution, and each subsequent layer must be batch-normalized. When the Ghost bottleneck is completed, it is used to build GhostNet. The first layer of the Ghost bottleneck requires a standard convolutional layer, and a series of Ghost bottlenecks are added to increase the number of channels. Finally, the feature map is converted into a dimensional feature vector by averaging the pool for the final recognition. Its structure diagram is shown in Fig. 2. The data of various electrical load characteristics of the LV station users are fed into the GhostNet model to identify multiple clustering centers obtained in the clustering operation.



(a) Convolutional layer (b) Ghost module

Fig. 2. (Color online) Ghost module.

## 4. Algorithm Flow

Figure 3 shows the algorithm flow.

## 5. Analysis Results of Simulation Experiment

### 5.1 Evaluation indicators

In this experiment, the clustering results were evaluated using three indicators: FM index (FMI), adjusted Rand index (ARI), and adjusted mutual information (AMI). C was set as the sample true label and C* was the result of clustering.

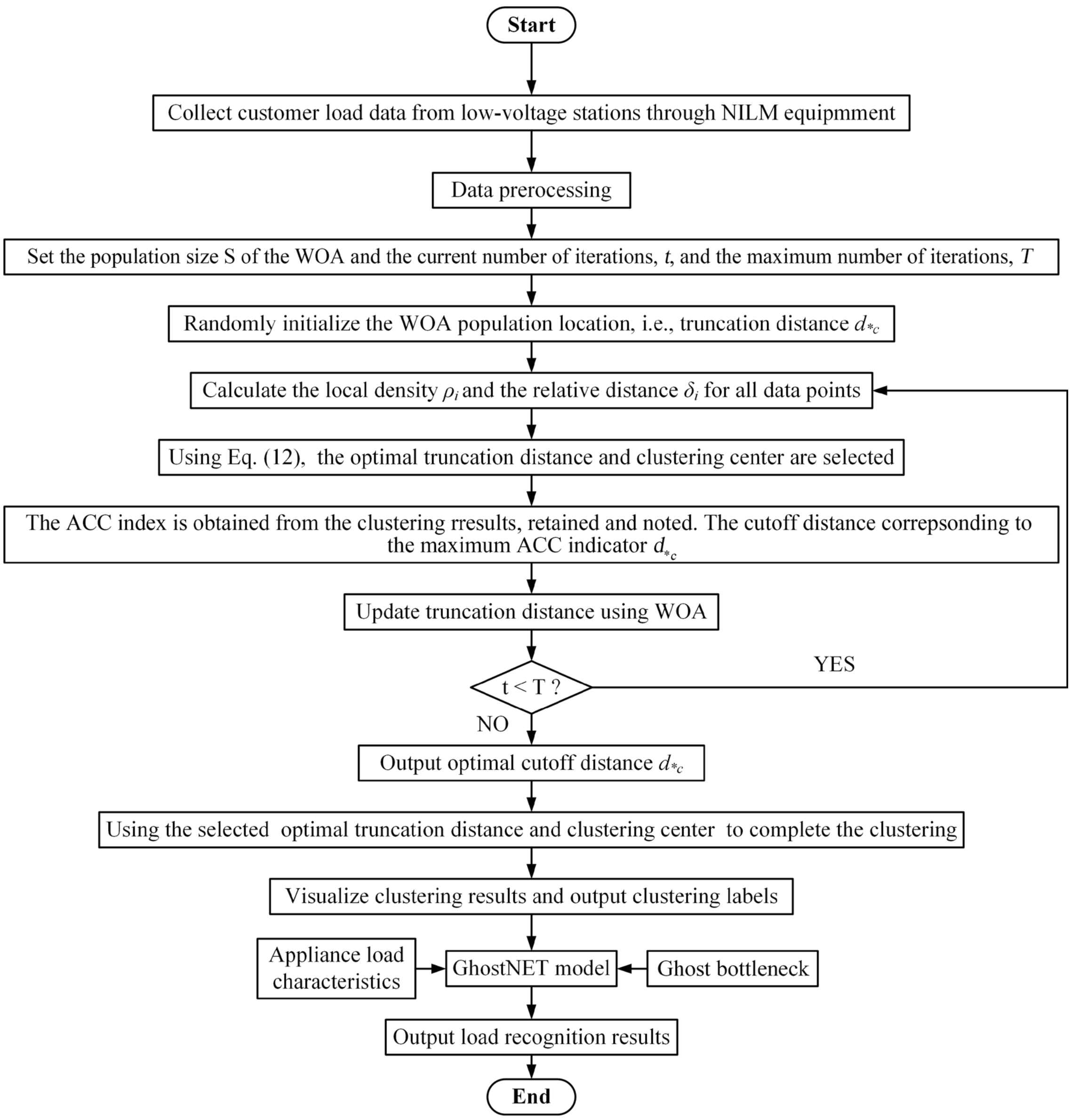



Fig. 3. Algorithm flow chart.

### 5.2 Results

The algorithm programming tool used was MATLAB R2020a, the operating system was Win10 64×, the memory was 16G, and the graphics card was NVIDIA 1060. The experimental data were the load data of 200 households in a district of a city in a province of China for one month, and six typical appliances were selected from all household appliances for identification: refrigerator, TV, laptop, microwave oven, induction cooker, and water heater.

In the experimental stage, the DPC, *k*-means, Euclidean clustering, and WOA–ADPC algorithm of this study were used and the results are shown in Table 1 and Figs. 4–6.

### 5.3 Analysis of results

Table 1 shows the clustering indexes of WOA–ADPC and other algorithms for the six appliances. Figures 4–6 show the three indexes of the four algorithms. DPC is slightly inferior to

Table 1
Results of WOA–ADPC and other algorithms.

| | WOA–ADPC | | | DPC | | |
|---|---|---|---|---|---|---|
| | FMI | ARI | AMI | FMI | ARI | AMI |
| Refrigerator | 0.987 | 0.925 | 0.931 | 0.851 | 0.890 | 0.899 |
| TV | 0.972 | 0.980 | 0.989 | 0.804 | 0.817 | 0.887 |
| Laptop | 0.917 | 0.917 | 0.901 | 0.800 | 0.802 | 0.745 |
| Microwave oven | 0.936 | 0.956 | 0.967 | 0.831 | 0.745 | 0.804 |
| Induction cooker | 0.934 | 0.911 | 0.945 | 0.759 | 0.871 | 0.865 |
| Water heater | 1.000 | 1.000 | 1.000 | 0.809 | 0.879 | 0.900 |
| | *k*-means | | | Euclidean clustering | | |
| | FMI | ARI | AMI | FMI | ARI | AMI |
| Refrigerator | 0.421 | 0.559 | 0.603 | 0.245 | 0.304 | 0.337 |
| TV | 0.431 | 0.478 | 0.432 | 0.207 | 0.214 | 0.228 |
| Laptop | 0.481 | 0.459 | 0.319 | 0.203 | 0.198 | 0.178 |
| Microwave oven | 0.402 | 0.420 | 0.397 | 0.167 | 0.188 | 0.201 |
| Induction cooker | 0.398 | 0.342 | 0.390 | 0.177 | 0.189 | 0.197 |
| Water heater | 0.489 | 0.475 | 0.420 | 0.296 | 0.303 | 0.316 |

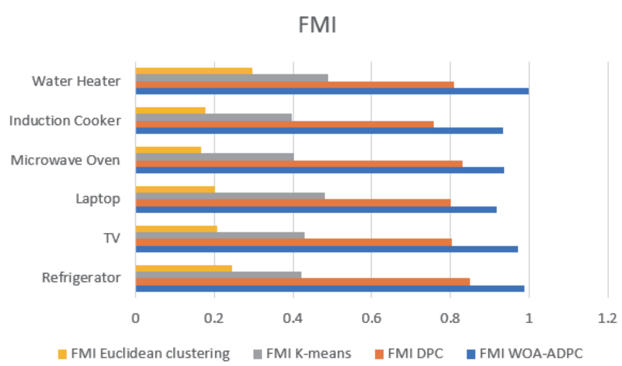



Fig. 4. (Color online) FMI results.

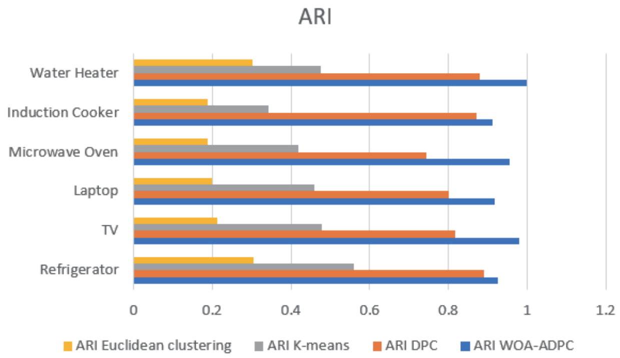



Fig. 5. (Color online) ARI results.

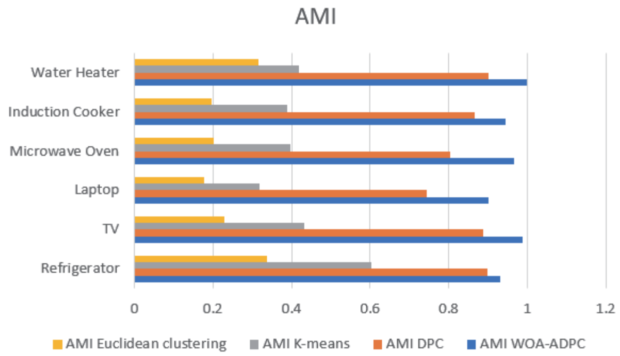



Fig. 6. (Color online) AMI results.

WOA–ADPC, with each indicator value being between 0.7 and 0.9. *k*-means differs significantly from that of DPC, with indicator values between 0.3 and 0.55. European-style clustering results are weaker than *k*-means, with indicator values being between 0.2 and 0.3. In summary, the clustering effect of our WOA–ADPC algorithm is better than those of the other three algorithms.

## 6. Conclusions

In this study, our focus was on the load identification of LV station users at the 10 kV voltage level. Specifically, we acquired user load data through a non-intrusive load monitoring (NILM). We used the improved ADPC algorithm combined with the WOA to cluster the discrete load data of multiple users and obtained results with different clustering centers. Considering the characteristics of the 10 kV voltage level, we made corresponding adjustments and optimizations in the data processing and analysis stage. These results were then input into the GhostNet model, and each clustering center was identified on the basis of the load characteristics of the appliances. By associating the appliances with their corresponding cluster centers, the load identification of 10 kV LV station users was accomplished.

Through simulation experiments, we found that our WOA–ADPC algorithm outperformed traditional DPC, *k*-means, and Euclidean distance clustering algorithms in terms of accuracy and clustering results, and it could better adapt to the load characteristics at the 10 kV voltage level. The WOA–ADPC algorithm also addressed the limitations of manual selection of clustering centers and achieved good results in load identification at the 10 kV voltage level. The next step is to study the application of the WOA–ADPC algorithm in handling high-dimensional data clustering issues at the 10 kV voltage level to further improve the accuracy and efficiency of load identification.

## Acknowledgments

This work was supported by the Science and Technology Project of China Southern Power Grid Co., Ltd. under Grant YNKJXM20230143.

## References

1 L Kong, Y. H. Luo, and D. S. Yang: 2017 1st IEEE Int. Conf. Energy Internet (ICEI 2017) 268. https://doi.org/10.1109/ICEI.2017.54
2 D. S. Yang, L. Kong, B. Hu, and T. Yuan: Autom. Electr. Power Syst. **41** (2017) 66. http://dx.doi.org/10.7500/AEPS20170516002
3 M. J. Du, S. F. Ding, and H. J. Jia: Knowledge-Based Syst. **99** (2016) 135. https://doi.org/10.1016/j.knosys.2016.02.001
4 M. Ester, H. Kregel, J. Sander, and X. Xu: 2nd Int. Conf. Knowledge Discovery and Data Mining (1996).
5 A. K. Jain: Pattern Recognit. Lett. **31** (2010) 651. https://doi.org/10.1016/j.patrec.2009.09.011
6 A. Saxena, M. Prasad, A. Gupta, N. Bharill, O. P. Patel, A. Tiwari, E. M. Joo, D. Weipin, and L. C. Teng: Neurocomputing **267** (2017) 664. https://doi.org/10.1016/j.neucom.2017.06.053 106–116. https://sci-hub.se/10.16749/j.cnki.jecjtu.20230209.006
7 Y. J. Xie, H. C. Gao, and W. X. Xie: China Sci.: Inf. Sci. **46** (2016) 258.
8 S. Wang, X. J. Xing, and T. Liu: J. East China Jiaotong University **40** (2023) 106. https://doi.org/10.16749/j.cnki.jecjtu.20230209.006
9 C. L. Ma, H. Shan, and T. Ma: Comput. Sci. **43** (2016) 255.
10 B. Wu, H. L. Lu, and H. J. Jiang: Comp. Appl. **40** (2020) 1654.
11 C. S. Liu and Q. L. Xu: Comp. Engi. Appl. **54** (2018) 153.
12 J. H. Wang, J. J. Li, J. S. Li, and W. D. Lai: Computer Engineering and Applications **55** (2019) 122.
13 X. Z. Qian and H. Jin: Comput. Sci. Explor. **14** (2020) 712. https://doi.org/10.3778/j.issn.1673-9418.1902022
14 S. Mirjalili and A. Lewis: Adv. Eng. Software **95** (2016) 51. https://doi.org/10.1016/j.advengsoft.2016.01.008
15 J. J. Liu, C. W. Qu, Y. F. Tang, and J. L. Li: Sci. Rep. **13** (2023) 3783. https://doi.org/10.1038/s41598-023-30862
16 Z. Z. Guo, P. Wang, Y. F. Ma, Q. Wang, and C. Q. Gong: Microelectron. Comput. **34** (2017) 20.
17 D. L. Chu, H. Chen, and X. G. Wang: J. Electron. **47** (2022) 992.
18 M. L. Zhang, Q. X. Wu, H. L. Chen, A. A. Heridari, Z. N. Cai, J. R. Li, E. M. Abderlrahim, and R. F. Mansour: Biomed. Signal Process. Control **83** (2023) 104638. https://doi.org/10.1016/j.bspc.2023.104638
19 Y. J. Cao and Y. X. Gao: Comput. Eng. **48** (2022) 310. https://sci-hub.se/10.19678/j.issn.1000-3428.0059966
20 C. J. Liu, X. P. Duan, and B. W. Xie: Laser Tech. **46** (2022) 239.
21 C. Y. Cheng, X. J. Wu, and T. Y. Xu: Pattern Recognit. Artif. Intell. **34** (2021) 1028. https://doi.org/10.16451/j.cnki.issn1003-6059.202111006
22 R. X. Cao and T. F Wang: IET Image Process. **17** (2023) 291. https://doi.org/10.1049/ipr2.12634
23 H. T. Zheng, H. Liu, Q. Wei, and H. Xie: Wireless Commun. Mobile Comput. (2022) 1. https://doi.org/10.1155/2022/5155970

## About the Authors

**Qiang Fu** received his bachelor degree in electrical engineering and automation from Kunming University of Science and Technology, Kunming China. He currently works at the Lijiang Power Supply Bureau of Yunnan Power Grid Co., Ltd. His research interests include power marketing and metering electricity. (616054972@qq.com)

**Xiaohua Yang** was born in 1988 in Jingdong County, Yunnan Province. He graduated from Yunnan University in 2011 with a major in electronic information and is currently an engineer and senior technician at the Metering Center (Power Load Control Technology Center) of Yunnan Power Grid Co. His main research directions are smart electricity usage and marketing metering. (yangxiaohua01@im.yn.csg)

**Jianyu Ren** was born in 1991 in Anning City, Yunnan Providence. He graduated from Kunming University of Science and Technology with a Electrical Engineering and automation and is currently an engineer at the Metering Center (Power Load Control Technology Center) of Yunnan Power Grid Co. His main research directions are smart electricity usage and marketing metering. (871050568@qq.com)

**Yonghui Zhao** was born in 1987 in Qujing City, Yunnan Province. He graduated from North China Electric Power University in 2013, majoring in control, and is now the manager of the Metering Center of Yunnan Power Grid Co. His main research direction is electrical energy metering.
(676808913@qq.com)

**Hao Yang** was born in 1990 in Qujing City, Yunnan Province. He graduated from North China Electric Power University majoring in Electrical Engineering and Automation, His main research directions are smart electricity usage and marketing metering. (yanghao05@im.yn.csg)

**Shibo Pan** is currently a master's student at the Civil Aviation College of Kunming University of Science and Technology. (1027949014@qq.com)