AN IMPROVED SPECTRAL CLUSTERING ALGORITHM FOR LARGE-SCALE WIND FARM POWER PREDICTION

Baohua Qiang,* Tian Zhao,* Wu Xie,* Hong Zheng,* Haoning Sun,* and Jinlong Chen*

Abstract

Aiming at reaching the balance between calculation efficiency and power prediction accuracy of wind farms, two improved spectral clustering (SC) algorithms and their application framework are proposed. For classical k-way Ng–Jordan–Weiss SC, the clustering sample space is composed of k eigenvectors, which may lose part of structural information and may not reach accurate clustering results. To improve the accuracy and stability, we proposed to cluster with feature expansion and the Cuckoo Search (CS) algorithm. We extended the clustering eigenspace from k eigenvectors to $2k$ to improve the clustering accuracy. To avoid following into local optimum while extending the eigenspace, the CS algorithm was introduced to search for better initial points instead of the random choice method. To apply the proposed algorithm for wind power prediction, wind turbines with similar wind regime were designated to the same group using the proposed SC algorithm. The power prediction model was established for each wind turbine group, and the output power of the entire wind farm was obtained by superposition. Experimental results indicated that the clustering accuracy is improved and the results of multiple clustering hold steady, which meets the requirement of accurate and timely prediction of wind farm power.

Key Words

Spectral clustering, power prediction, Cuckoo Search algorithm, Levy flight

1. Introduction

Wind farm power prediction technology is one of the crucial means to solve the impact of large-scale wind power grid generation on the safety, stability and economic operation of smart grids [1]. The mainstream approach utilizes the wind regime of typical locations to predict the output power of an entire wind farm [2]. For large-scale wind farms, this strategy cannot guarantee prediction accuracy. But if the prediction is based on an individual wind turbine, the calculation time will be too long, which cannot meet the real-time prediction requirements of smart grids. On the other hand, if the similarity between different wind turbines can be identified and thus the entire wind farm can be divided into different groups of wind turbines to predict in batches, the prediction accuracy can be improved, and the computational efficiency can be guaranteed. Therefore, the study of an efficient and stable clustering algorithm is the key issue of power prediction for a large-scale wind farm.

Data clustering represents a valuable data mining tool in the fields of computer vision [3]–[5], text mining [6], [7], bioinformatics mining [8], [9], *etc.* It explores the potential value of data by classifying samples into different groups based on their degree of association [10]. For non-linear separable problems, spectral clustering (SC) [11]–[13] is among the most effective clustering methods. For example, experiments in [14] demonstrated that the forecasting model based on the SC algorithm shows a higher accuracy on wind power prediction than those models combined with other clustering algorithms. The SC algorithm was first derived from the graph spectral partitioning theory [15], which aims at solving the cluster partitioning problem using the optimal partitioning theory of graphs. According to the partitioning criterion of graph theory, SC can be classified as iterative SC [16], [17] and k-way SC [18], [19]. In 1998, Perona *et al.* [16] proposed to cluster with the eigenvector x_1 corresponding to the largest eigenvalue of the affinity matrix. Shi *et al.* [17] presented a new solution for graph partitioning and came up with a two-way partitioning normalized-cut objective function. The Ng–Jordan–Weiss (NJW) method proposed by Ng *et al.* [18] is a classical iterative SC algorithm, which calculates the sample affinity matrix, converts it into a Laplacian matrix, computes the Laplacian matrix eigenvalues, and then clusters with the eigenvectors corresponding to the k largest eigenvalues. The Markov chain clustering algorithm proposed by Meila [19] has similar algorithmic steps with the NJW algorithm, but its eigenspace is composed of eigenvectors obtained by the random walk matrix.

Nowadays, researches on SC mainly focus on the construction of the affinity matrix, selection of eigenvectors, determination of the cluster number, and their

* School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin, China; e-mail: qiangbh@guet.edu.cn, {781609321, 1094678426, 1094678426}@qq.com, guetxie@163.com, zhenghong@guet.edu.cn
Corresponding author: Wu Xie
Recommended by Prof. Anmin Zhu
(DOI: 10.2316/J.2019.206-0260)

applications. In these research fields, the selection of eigenvectors plays a crucial role in the division of clusters. It was proved in [20] that it is not always possible to select a group of eigenvectors suitable for clustering all data samples. Even if such a solution set exists, it is hard to obtain with little prior knowledge. The classical NJW algorithm [18] performs clustering operation using the eigenvectors corresponding to the k largest eigenvalues. Instead of using eigenvectors, Sun *et al.* [21] searched the feature dimension of the principal component by swarm intelligent algorithm and took the vector subset reflecting the relationship between data samples as the clustering space vectors. Zhao *et al.* [22] calculated the entropy of the eigenvectors and selected the cluster space vectors based on the sorted entropy values (ESBER_D). They also demonstrated that the cluster resulting from k-dimensional eigenspace is not necessarily the optimal clusters and that there are other combinations of eigenvectors which can achieve better results. Rebagliati *et al.* [23] indicated that the number of eigenvectors can be determined by referring to the difference between the eigenvalues. They argued that, to some extent, the eigenvalues can reflect the corresponding eigenvectors' ability to classify clusters. In the landmark-based SC [11], the original data points are represented by the linear combinations of those landmarks of representative data points. Despite intense research, selecting the optimal eigenvector for SC is still open.

According to the above discussion on the selection of eigenvectors, it reveals that clustering using eigenvectors corresponding to the k largest eigenvalues may lose some feature information. We considered that using more eigenvectors to form the cluster eigenspace may further improve the accuracy. Therefore, we extended the dimension of eigenspace of the NJW algorithm from k-dimension to $2k$-dimension (*i.e.* cluster with the eigenvectors corresponding to the first $2k$ largest eigenvalues). To avoid cluster results fluctuation caused by falling into local optimum, the Cuckoo Search (CS) algorithm [24] was introduced, and the sum error square was employed as the fitness function to select high-quality initialization centres. Hence, an improved NJW algorithm based on feature expansion and CS was proposed. The clustering accuracy and stability of the new model were validated on four groups of experimental data. Owing to the high accuracy of the SC method in predicting wind power [14], we also set out to apply the proposed algorithm on grouping the wind turbines of large-scale wind farms and then predict the power of each group. The output power of the entire farm is obtained by the sum power of every group. The smoothing effect of multiple groups of wind turbines can improve the accuracy of power prediction, and the batch prediction strategy can ensure the computational efficiency so as to provide more timely power prediction for power system scheduling.

2. Basic Concepts

2.1 Spectral Clustering

SC is a clustering algorithm based on graph theory, which transforms the clustering problem into the optimal partitioning problem of graphs. The concepts involved in SC are as follows:

1. Graph representation
   $G(V, E)$ denotes an undirected graph, where the vertex set is represented by $V = \{v_1, v_2, \ldots, v_n\}$, the edge set is denoted by E, the weigh between v_i and v_j is expressed by w_{ij}, and for an undirected graph:

$$w_{ij} = w_{ji}, \quad w_{ii} = 0, \quad w_{ij} \geq 0 \tag{1}$$

2. Graph partition
   Graph partition refers to dividing a graph into a number of completely separated subgraphs, that is,

$$G_1 \cup \ldots \cup G_k = G \tag{2}$$

$$G_1 \cap G_k = \emptyset \tag{3}$$

3. Loss function
   The loss function is defined as the weighted sum of the edges which are truncated between subgraphs:

$$\mathrm{Cut}(G_1, G_2) = \sum_{i \in G_1, j \in G_2} w_{ij} \tag{4}$$

   where the criterion for a division is that there are high similarity within subgraphs and low similarity between subgraphs.
4. Laplacian matrix
   Let $G(V, E)$ (suppose G has n points) be divided into two subgraphs, $G1$ and $G2$, and the n-dimension vector $q = [q_1, q_2, \ldots, q_n]$ represents the partition scheme.

$$q_i = \begin{cases} c_1 & i \in G_1 \\ c_2 & i \in G_2 \end{cases}, \quad q = [c_1, c_1, c_1, c_2, c_2, c_2] \tag{5}$$

   The corresponding loss function is

$$\mathrm{Cut}(G_1, G_2) = \sum_{i \in G_1, j \in G_2} \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} (q_i - q_j)^2}{2(c_1 - c_2)^2} \tag{6}$$

where

$$\begin{aligned} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} (q_i - q_j)^2 &= -\sum_{i=1}^{n} \sum_{j=1}^{n} 2 w_{ij} q_i q_j + \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} (q_i^2 + q_j^2) \\ &= 2 q^T (D - W) q \end{aligned} \tag{7}$$

where D is a diagonal matrix:

$$D_{ii} = \sum_{i=1}^{n} w_{ij} \tag{8}$$

The Laplacian matrix is defined as

$$L = D - W \tag{9}$$

where W is the weight matrix (also called the adjacency matrix), and D is the similarity degree matrix.

$$\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}(q_i - q_j)^2 = 2q^T Lq \tag{10}$$

$$q^T Lq = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}(q_i - q_j)^2 \geq 0 \tag{11}$$

L is a positive semi-definite matrix. The minimum eigenvalue is zero, and its eigenvector is the unit vector.

Based on the above definitions, the loss function is expressed as

$$\mathrm{Cut}(G_1, G_2) = \sum_{i\in G_1, j\in G_2} \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}(q_i - q_j)^2}{2(c_1 - c_2)^2} \tag{12}$$

$$\mathrm{Cut}(G_1, G_2) = \frac{q^T Lq}{(c_1 - c_2)^2} \tag{13}$$

Thus, the graph partition problem is transformed into the $q^T Lq$ conditional minimum problem.

2.2 k-Way NJW Spectral Clustering Algorithm

The NJW algorithm [18] is the classical k-way SC algorithm. It first constructs the Laplacian matrix, then forms the eigenspace by the eigenvectors corresponding to the k largest eigenvalues, and finally divides the sample within eigenspace using the k-means or other clustering algorithms. The main steps of the NJW algorithm are as follows:

Algorithm 1. k-Way NJW SC algorithm

Step 1. Construct the affinity matrix $A \in n \times n$,A_{ij} is calculated as

$$A_{ij} = \exp\left(\frac{-|d_{ij}|^2}{2\sigma^2}\right), \quad i \neq j, \quad A_{ii} = 0 \tag{14}$$

Step 2. Calculate the Laplacian matrix L as

$$L = D^{-(1/2)} A D^{-(1/2)} \tag{15}$$

where D is the diagonal matrix with the diagonal element D_{ij}:

$$D_{ii} = \sum_{j=1}^{n} A_{ij} \tag{16}$$

Step 3. Compute the eigenvectors $(x_1 x_2 \ldots x_k)$ corresponding to the k largest eigenvalues, forming the matrix $X = [x_1 x_2 \ldots x_k] \in n \times k$

Step 4. Normalize the row vectors of matrix X to get matrix Y, and Y_{ij} is calculated as

$$Y_{ij} = \frac{X_{ij}}{\left(\sum_j X_{ij}^2\right)^{1/2}} \tag{17}$$

Step 5. Each row in matrix Y is treated as a data point of R^k, which divides Y into k classes by the k-means or other clustering algorithms.

Step 6. Finally, when and only when the ith row of matrix Y is classified to the jth cluster, the original data point, s_i, is designated to the jth cluster.

The NJW algorithm works by mapping the original data distribution to another space dimension, so as to make the data distribution in the latter dimension tighter than the origin to improve the clustering capacity of each feature dimension. The mechanism lies in that each feature dimension possesses the ability to classify clusters, that is, the ability to represent a certain feature. But the impact of different features on partitioning is not the same. Therefore, clustering only with the eigenvectors corresponding to the k largest eigenvalues will lose the feature information of some vector dimensions, which reduces the clustering accuracy.

2.3 The Cuckoo Search Algorithm

The CS algorithm [24] is an optimization model based on the parasitic breeding behaviour of some cuckoo species and Levy flight behaviour of some birds. It has the characteristics of being insensitive to parameter changes, being able to jump out of local optimum, and being easy to integrate into other algorithms.

To simulate the behaviour of cuckoo nesting and spawning, the following three states need to be assumed:

1. The cuckoo chooses the location for nesting at random and lays one egg at a time.
2. The nest locations selected by different birds are different, and the optimal location is retained.
3. The number of nests is fixed, and the probability of boarding eggs being found is P_a, $P_a \in [0, 1]$.

Based on the above idealized assumption, the formula for location updating of CS is as follows:

$$x_i^{t+1} = x_i^t + \alpha \oplus \text{Lévy}(\lambda) \tag{18}$$

The CS is a swarm intelligence algorithm, which simulates the message passing and group collaborating among biological individuals and can effectively solve many global optimization problems. This algorithm is simple and effective, has strong global search ability and strong robustness, and can provide effective solutions for complex optimization problems. Combining swarm intelligence algorithm with clustering algorithm is not only feasible but also can improve the clustering performance.

3. Improved NJW Algorithm based on Feature Expansion and Cuckoo Search

3.1 The NJW Algorithm based on Feature Expansion

For the eigenvectors of Laplacian matrix obtained by the NJW algorithm, there is no standard practice for determining the dimension and combination of eigenvectors for clusters. The eigenvectors corresponding to larger eigenvalues are more crucial for spectral space representation, and eigenvectors of each dimension have some ability to divide clusters. Therefore, using the eigenvectors corresponding to the largest $2k$ eigenvalues to form the cluster eigenspace may improve the clustering performance. On the other hand, eigenvectors of each dimension also produce information that interferes with clustering. Therefore, different combinations of multidimensional vectors will have different effects. When the dimension of eigenspace is increased to $2k$, compared with the classical NJW algorithm, the clustering accuracy may rise or decrease. The result fluctuation of multiple clustering is higher than that of the classical NJW algorithm. Expanding the eigenspace from k to $2k$ dimension may introduce some dimension eigenvectors with poor ability to classify clusters, which increases the irrelevancy between data and makes data points more scattered than those in low-dimensional space.

At the same time, the random selection initialization strategy adopted by the NJW clustering algorithm increases the probability of falling into a local optimum, which leads to the fluctuation of multiple clustering. To reduce the interference information introduced by expanding eigenvectors and to maintain clustering stability, we solve the above problems by selecting comparatively better initialization centres.

3.2 Spectral Clustering Algorithm based on Feature Expansion and Cuckoo Search

In this paper, we introduce the CS algorithm to find the initialization centre points, which is based on the calculation of a fitness function to find a high-quality location. The sum of square error is used as the fitness function in the CS algorithm, which is defined as

$$F = \sum_{j=1}^{k} \sum_{i=1}^{n_j} \left| x_i^{(j)} - c_j \right|^2 \tag{19}$$

where x_i is the sample data point, and $c_j (j = 1, 2, 3, \ldots, k)$ represent the cluster centres.

By introducing the CS algorithm and taking those data points with the minimum sum of square error as the initialization centre points to the improved NJW algorithm, the accuracy instability caused by the random selection initialization strategy is reduced. The new algorithm is called CS_$2K$_NJW, and the calculation process is shown in Algorithm 2.

Algorithm 2. Improved NJW algorithm based on feature expansion and CS:

Input: The whole data sample

Output: The clustering results with centres

Step 1. Set the initialization parameters and the cluster number k;

Step 2. According to formulas (14)–(17), calculate the Laplacian matrix L and the normalized matrix Y of the eigenvectors corresponding to the largest $2k$ eigenvalues;

Step 3. According to formula (19), take those n data points with the minimum sum of square error as the initial nest locations $Y_i^{(0)} = [Y_1^{(0)}, Y_2^{(0)}, \ldots, Y_n^{(0)}]^T$;

Step 4. Cluster the data sample with the n initial nest locations. As shown in formula (20), calculate the fitness value F of each position and retain the result of relatively small F value:

$$F_{\text{best}} = \min\{F_1, F_2, \ldots, F_n\} \tag{20}$$

Step 5. Calculate the new nest location according to the Levy flight strategy in CS algorithm;

Step 6. Classify the updated nest locations, compute the new fitness value of each nest location, compare the two generations of nest location according to their fitness values, and retain the relatively small F according to the following formula;

$$F_j^i = \min\left\{F_j^{i-1}, F_j^i\right\} \tag{21}$$

$$F_{\text{best}}^i = \min\left\{F_{\text{best}}^{i-1}, F_1^i, F_2^i, \ldots, F_n^i\right\} \tag{22}$$

Step 7. Generate a random number $r \in [0, 1]$ and compare it with probability P_a. If $r < P_a$, retain the results, otherwise, recalculate the location by the update formula;

Step 8. If the maximum iteration number or the stop condition is satisfied, keep the solution with the minimum fitness value and proceed with the following operation, otherwise, continue computing from Step 5;

Step 9. Carry out k-means clustering on the retained optimal nest locations, and finally output the clustering results.

4. Experimental Results and Analysis

4.1 Test Dataset

In this paper, experiments are conducted on three datasets from the University of California Irvine machine learning databases [25] and the dataset from literature [26]. The characteristics of the four datasets are shown in Table 1, where the Path-based dataset is two dimensional with its distribution shown in Fig. 1. It is observed that this dataset has the characteristic of the non-convex distribution.

Table 1
Basic Information about the Four Datasets

Name	Instances	Attributes	Clusters	Instances in Each Cluster
Wine	178	13	3	59/71/48
Iris	150	4	3	50/50/50
Seeds	210	7	3	70/70/70
Path-based	300	2	3	110/93/97

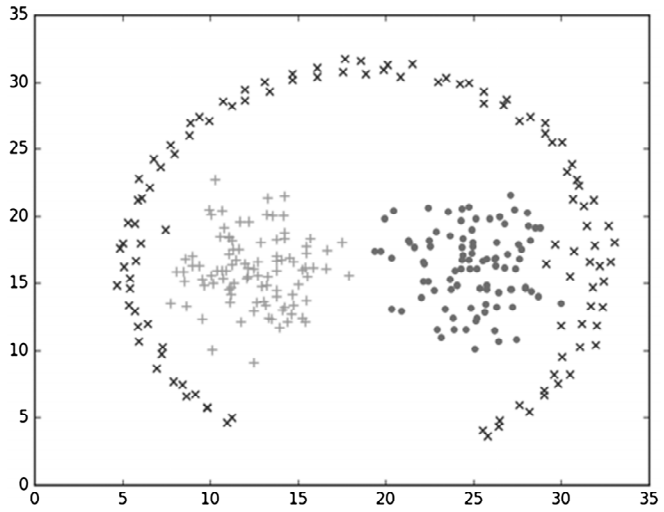


Figure 1. Distribution of the Path-based dataset.

4.2 Measurement

To evaluate the effectiveness and stability of the proposed algorithm, we process the above four groups of data for 20 times using the k-means, NJW, ESBER_D based on entropy sorting, and the proposed $2K$_NJW and CS_$2K$_NJW algorithms. Clustering accuracy [27] is adopted to evaluate their performances, which is defined as

$$\text{accrucy} = \frac{\sum_{i=1}^{n} \delta(y_i, \text{map}(c_i))}{n} \tag{23}$$

where n denotes the number of samples in the dataset, y_i marks the cluster to which a data sample belongs, c_i denotes the final clustering results, and $\delta(y_i, \text{map}(c_i))$ counts the number of correct clustering results. $\text{map}(\cdot)$ converts the resulting to the original cluster identifier. The clustering accuracy is measured by the maximum value of $\delta(y_i, \text{map}(c_i))$.

4.3 Parameter Setting

In the experiments, $\sigma = 1$, d_{ij} denotes the sample similarity based on the Euclidean distance. For the SC algorithm, P_a is fixed to 0.25. In (18), α is the step factor and is set to 1, $\oplus$ denotes dot product. Lévy(λ) represents Lévy random search path with the following distribution.

$$\text{Lévy}(\lambda) \sim u = t^{-\lambda}, \quad 1 < \lambda \leq 3 \tag{24}$$

It is usually difficult to compute the Lévy distribution by the probability density function. In 1994, Rosario Mantegna proposed a numerical method which computes the Lévy distribution by the following formulae [28]:

$$s = \frac{u}{|v|^{1/\beta}} \tag{25}$$

$$u \sim N(0, \sigma_u^2), \quad v \sim N(0, \sigma_v^2) \tag{26}$$

$$\sigma_u = \left\{ \frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma[(1+\beta)/2]\beta 2^{(\beta-1)/2}} \right\}^{1/\beta}, \sigma_v = 1 \tag{27}$$

Lévy(λ) is calculated from (24). All values of β in (25) and (27) are usually constant 1.5, and the result of σ_u is also constant 0.6966. From (25)–(27), it can be seen that s depends on u and v, both of which are random numbers obeying the normal distribution. The symbols of u and v can be positive or negative, and their values can be large or small, so that the step size and the direction of levy's flight are randomly changed, which strengthens the global search ability of the CS algorithm and is conducive to jumping out of local optima.

4.4 Experiment and Results

Figures 2–5 are 20 clustering results of the above five algorithms on four datasets. Tables 2–5 are the maximum, minimum, and average accuracy of 20 clustering on the dataset Wine, Iris, Seeds, and Path-based, respectively.

It is clearly observed from Figs. 2–5 that the k-means performs the worst on the stability of multiple clustering. For the classical NJW algorithm, the accuracy of 20 clustering on the wine, seeds, and Path-based datasets remains stable. For the Seeds dataset, though there is some fluctuation, most of the 20 clustering of the NJW algorithm hold the same accuracy. Compared with the results of NJW, the ESBER_D algorithm manifests different performances for different datasets, indicating that the quality of eigenspace directly influences the clustering results. For the proposed

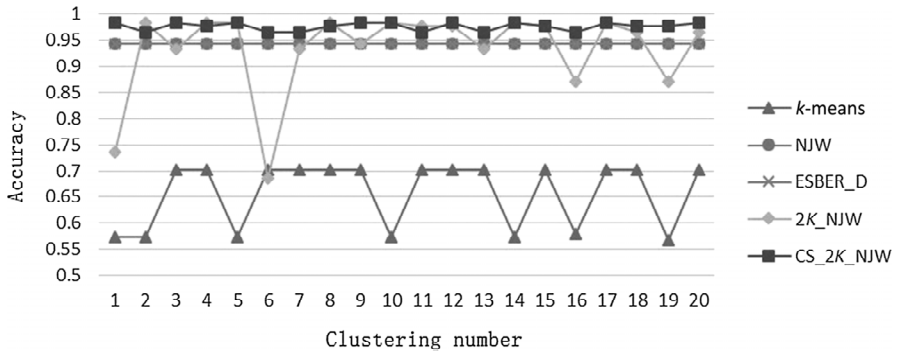


Figure 2. Twenty clustering results on the Wine dataset.

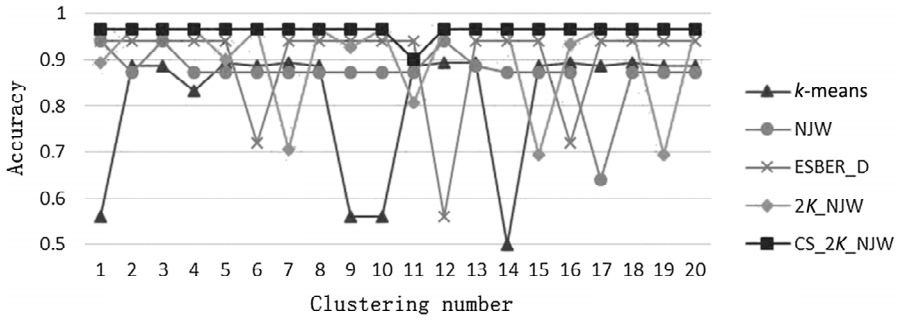


Figure 3. Twenty clustering results on the Iris dataset.

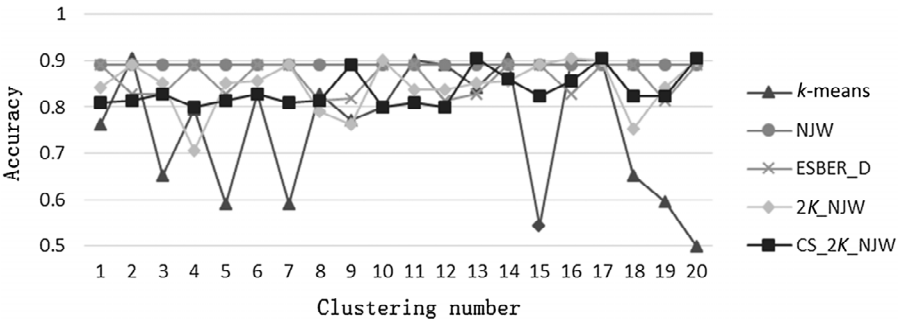


Figure 4. Twenty clustering results on the Seeds dataset.

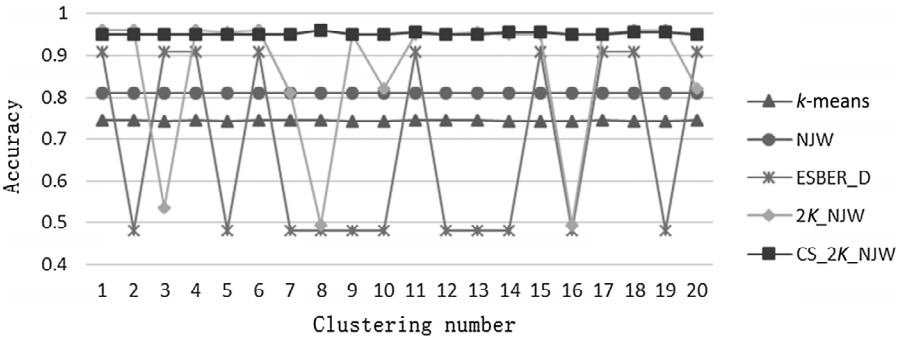


Figure 5. Twenty clustering results on the Path-based dataset.

Table 2
Clustering Accuracy on the Wine Dataset

	Acc_min	Acc_max	Acc_avg
k-Means	56.7	70.2	65.7
NJW	94.4	94.4	94.4
ESBER_D	94.4	94.4	94.4
$2K$_NJW	68.5	98.3	3.4
CS_$2K$_NJW	96.6	98.3	97.7

Table 3
Clustering Accuracy on the Iris Dataset

	Acc_min	Acc_max	Acc_avg
k-Means	50.0	89.3	81.7
NJW	64.0	94.0	87.2
ESBER_D	56.3	94.0	89.9
$2K$_NJW	69.3	96.7	90.7
CS_$2K$_NJW	90.0	96.7	96.3

Table 4
Clustering Accuracy on the Seeds Dataset

	Acc_min	Acc_max	Acc_avg
k-Means	50.0	90.4	75.8
NJW	89.0	89.0	89.0
ESBER_D	81.4	89.0	86.0
$2K$_NJW	70.4	90.5	84.6
CS_$2K$_NJW	80.0	90.5	83.6

Table 5
Clustering Accuracy on the Path-based Dataset

	Acc_min	Acc_max	Acc_avg
k-Means	74.3	74.7	74.5
NJW	81.0	81.0	81.0
ESBER_D	48.0	91.0	67.4
$2K$_NJW	49.3	96.0	86.7
CS_$2K$_NJW	95.0	96.0	95.2

$2K$_NJW algorithm, some results are significantly better than those of the NJW algorithm. It reveals that the information provided by additional k-dimension eigenvectors helps to accomplish an optimal partitioning to some extent. But the stability of 20 clustering of the proposed $2K$_NJW algorithm is very poor. This is because some combinations of the additional k eigenvectors exert a positive impact, but other combinations have a negative effect on the clustering. Higher eigenspace dimension increases the dispersion degree of data points, making the algorithm become more sensitive to initial centres and falling into the local optimal solution. As for the CS_$2K$_NJW algorithm, though the clustering results on the Seeds dataset are not so good, it almost provides the highest accuracy of all 20 clustering on the Wine, Iris, and Path-based datasets. There is one thing required to be pointed out that the proposed CS_$2K$_NJW algorithm has the ability to cluster on a non-convex distribution dataset as shown in Fig. 1.

It can be observed from Table 2–5 that the maximum accuracy of both the $2K$_NJW and the CS_$2K$_NJW algorithm are higher than that of the NJW and ESBER_D algorithm. This complies with the conclusion of [23] that the number of eigenvectors for the highest accuracy clustering results is not necessarily k. The minimum accuracy of the $2K$_NJW algorithm is also very low. The reason is that the information provided by each dimension is different, and too many feature dimensions also cause interference. With the purpose of minimizing the sum of variances of all clusters and taking full advantage of the ability of each vector dimension to divide class clusters, the proposed CS_$2K$_NJW algorithm combines the CS algorithm with the improved NJW SC with $2k$ eigenvectors. Due to initializing with high-quality cluster centres and extra information provided by additional k-dimension eigenvectors, the proposed CS_$2K$_NJW algorithm proves high performance on clustering stability and accuracy. As shown in Tables 2, 3, and 5, the average accuracy of 20 clustering on the Wine, Iris, and Path-based datasets of the proposed CS_$2K$_NJW algorithm is higher than those of all other algorithms.

As shown in Table 4, the highest clustering accuracy obtained by the CS_$2K$_NJW algorithm on the Seeds dataset is no less than those of the other algorithms, but the average accuracy is not so good. The main reason is as follows. In this paper, the square error of every single cluster is not addressed in the fitness; it may occur that the square error of some single cluster is small due to the small sample number, resulting in a small fitness value. The experimental results on the Seeds dataset reveal that smaller fitness values correspond to worse clustering results as shown in Fig. 6. Though the swarm intelligence

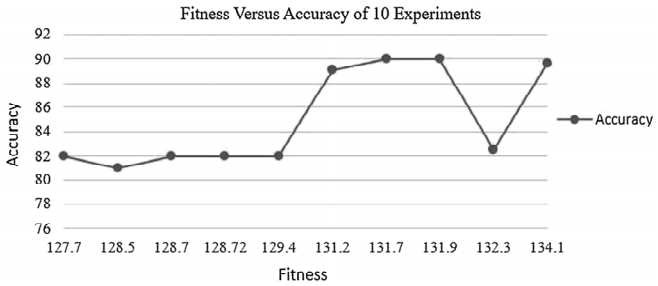


Figure 6. Fitness versus accuracy of 10 experiments.

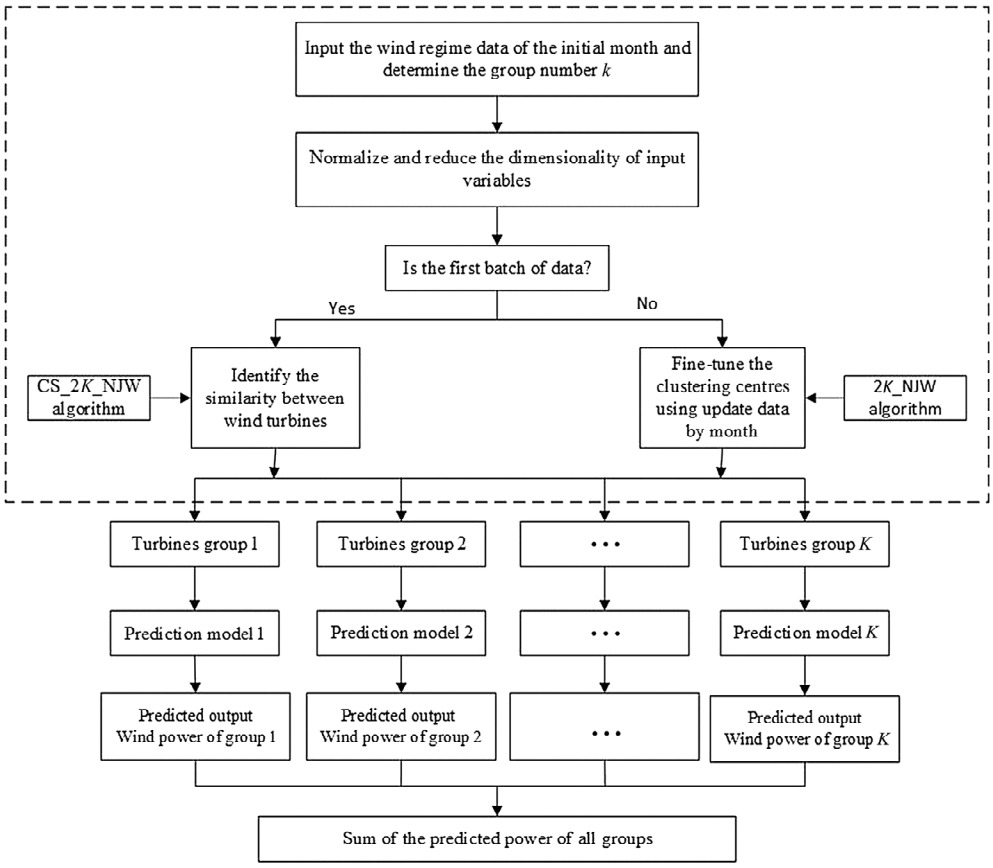


Figure 7. Framework of large-scale wind farms power prediction.

algorithm SC can converge to an optimal solution, due to the lack of evaluation of each cluster, the initial centres determined by the SC may be a locally optimal solution for clustering, resulting in bad final clustering results.

Although the computation complexity is increased, the above experimental results and analysis demonstrate that, by feature expansion and initialization with the CS algorithm, the overall performance on clustering accuracy and stability of the proposed CS_2K_NJW algorithm is higher than that of the k-means, classical NJW, and ESBER_D algorithms.

4.5 Application of the CS_2K_NJW Algorithm in Real Data

As real wind power data are still being collected, we can only propose to apply the proposed two algorithms on power prediction of large-scale wind farms. First, input the wind regime data of the initial month and the wind farm group number k. Second, the wind turbines with similar wind regime, and force outputs are assigned to the same group by the CS_2K_NJW algorithm. Finally, the power prediction model is established for each wind turbine group, and the output power of the entire wind farm is obtained by superposition. The wind regime data of any follow-up month are employed to fine-tune the previous turbine groups using the proposed 2K_NJW algorithm. The framework for wind turbine grouping and power prediction modelling is presented in Fig. 7. The specific steps include the following:

1. Collect wind farm data and determine the input variables for the wind turbine grouping model.
2. Settle down the group number k.
3. Normalize and reduce the dimensionality of input variables.
4. Identify the similarity between wind turbines by the CS_2K_NJW algorithm and group them to form k clusters.
5. Establish power prediction model for each wind turbine group. The predicted power of the entire wind farm is the sum of the predicted power of each group.

5. Conclusion

For large-scale wind farms power forecasting, it is difficult to guarantee the accuracy and efficiency of prediction simultaneously. To solve this problem, a SC algorithm based on feature extension and CS and its application framework is proposed in this paper. Through multiple SC experiments of extending the eigenspace from k to $2k$, it reveals that, compared with the classical NJW algorithm, the clustering accuracy can frequently fluctuate. We also find out that the random choice initialization strategy is responsible for the fluctuation. Therefore, we introduce the SC algorithm to the improved NJW algorithm to search for better initial points, which can effectively avoid trapping

in local optimal and improve the clustering stability. Experimental results on four datasets reveal that, compared with the baseline methods, the proposed CS_$2K$_NJW algorithm can achieve more accurate clustering results as well as better stability of multi-clustering, which may meet the requirement of accurate and real-time forecasting of the wind power output.

As real wind power data are still being gathered, we can only propose a framework for wind turbine grouping and power prediction. Next, we will evaluate the prediction performance of the proposed algorithm in power prediction of large-scale wind farms.

Acknowledgement

This work is partially supported by Guangxi Science and Technology Planning Project (Nos. AB17195053, AD18281002, AB18126063, and AB18126053), the National Marine Technology Program for Public Welfare (No. 201505002), the National Natural Science Foundation of China under Grant Nos. 61762025 and 61662014, Guangxi Natural Science Foundation under Grant Nos. 2017GXNSFAA198226 and 2018GXNSFAA294052, Guangxi Key Laboratory of Trusted Software (kx201510 and kx201413), Guangxi Colleges and Universities Key Laboratory of Cloud Computing and Complex System (Nos. 14106 and 15204), the Innovation Project of GUET Graduate Education (Nos. 2017YJCX52 and 2018YJCX42), Guangxi Cooperative Innovation Center of Cloud Computing and Big Data (Nos. YD16E01, YD16E04, YD1703, YD1712, YD1713, and YD1714), and Guangxi Colleges and Universities Key Laboratory of Intelligent Processing of Computer Image and Graphics (No. GIIP201603).

References

[1] L. Dong, L. Wang, S.F. Khahro, *et al.*, Wind power day-ahead prediction with cluster analysis of NWP, *Renewable & Sustainable Energy Reviews*, *60*, 2016, 1206–1212.

[2] Y. Lu, T. Zhang, Z. Zeng, *et al.*, An improved RBF neural network for short-term load forecast in smart grids, *International Conf. on Conceptual Structures*, Annecy, France, 2016, 1–6.

[3] R. Peng, H. Sun, L. Zanetti, *et al.*, Partitioning well-clustered graphs: Spectral clustering works!, *Conference on Learning Theory*, *46*(2), 2017, 1423–1455.

[4] Y. Yang, Y. Wang, X. Xue, *et al.*, A novel spectral clustering method with superpixels for image segmentation, *Optik*, *127*(1), 2016, 161–167.

[5] H. Zbib, S. Mouysset, S. Stute, *et al.*, Unsupervised spectral clustering for segmentation of dynamic PET images, *IEEE Transactions on Nuclear Science*, *62*(3), 2015, 840–850.

[6] I.S. Dhillon, Co-clustering documents and words using bipartite spectral graph partitioning, *Knowledge Discovery and Data Mining*, San Francisco, America, 2001, 269–274.

[7] V. Mijangos, G. Sierra, A. Montes, *et al.*, Sentence level matrix representation for document spectral clustering, *Pattern Recognition Letters*, *85*, 2017, 29–34.

[8] C.H. Ding, Unsupervised feature selection via two-way ordering in gene expression analysis, *Bioinformatics*, *19*(10), 2003, 1259–1266.

[9] W. Zang, Z. Jiang, L. Ren, *et al.*, Improved spectral clustering based on density combining DNA genetic algorithm, *International Journal of Pattern Recognition and Artificial Intelligence*, *31*(04), 2017, 1750010-1–1750010-23.

[10] F. Ding, J. Wang, J. Ge, *et al.*, Anomaly detection in large-scale trajectories using hybrid grid-based hierarchical clustering, *International Conference on Robotics and Automation*, *33*(5), 2018, 474–480.

[11] R. Langone and J.A. Suykens, Fast kernel spectral clustering, *Neurocomputing*, *268*, 2017, 27–33.

[12] A. Vora and S. Raman, Iterative spectral clustering for unsupervised object localization, *Pattern Recognition Letters*, *106*, 2018, 27–32.

[13] Y. Xu, Z. Zhuang, W. Li, *et al.*, Effective community division based on improved spectral clustering, *Neurocomputing*, *279*, 2017, 54–62.

[14] R. Yu, J. Gao, M. Yu, *et al.*, LSTM-EFG for wind power forecasting based on sequential correlation features, *Future Generation Computer Systems, 93*, 2019, 33–42.

[15] M. Fiedler, Algebraic connectivity of graphs, *Czechoslovak Mathematical Journal*, *23*(23), 1973, 298–305.

[16] P. Perona and W.T. Freeman, A factorization approach to grouping, *European Conference on Computer Vision*, Freiburg, German, 1998, 655–670.

[17] J. Shi and J. Malik, Normalized cuts and image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*(8), 2000, 888–905.

[18] A.Y. Ng, M.I. Jordan, Y. Weiss, *et al.*, On spectral clustering: Analysis and an algorithm, *Neural Information Processing Systems*, Vancouver, Canada, 2001, 849–856.

[19] M. Meila and J. Shi, Learning segmentation by random walks, *Neural Information Processing Systems*, Denver, America, 2000, 873–879.

[20] T.G. Dietterich, Machine learning research: Four current directions, *AI Magazine*, *18*(4), 1997, 97–136.

[21] Z. Sun, G. Bebis, R. Miller, *et al.*, Object detection using feature subset selection, *Pattern Recognition*, *37*(11), 2004, 2165–2176.

[22] F. Zhao, L. Jiao, H. Liu, *et al.*, Spectral clustering with eigenvector selection based on entropy ranking, *Neurocomputing*, *73*(10), 2010, 1704–1717.

[23] N. Rebagliati and A. Verri, Spectral clustering with more than K eigenvectors, *Neurocomputing*, *74*(9), 2011, 1391–1401.

[24] X. Yang and S. Deb, Cuckoo Search via Lévy flights, *Nature and Biologically Inspired Computing,* Coimbatore, India, 2009, 210–214.

[25] A. Asuncion and D.J. Newman, UCI Machine Learning Repository, 2007, http://www.ics.uci.edu/∼mlearn/MLRepository.html.

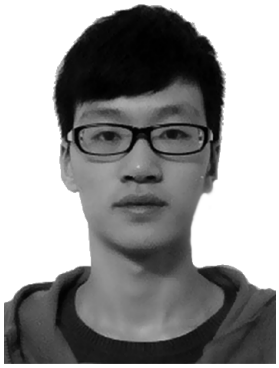
[26] H. Chang, D.Y. Yeung, Robust path-based spectral clustering, *Pattern Recognition*, *41*(1), 2008, 191–203.

[27] M. Wu and B. Scholkopf, A local learning approach for clustering, *Neural Information Processing Systems*, Vancouver, Canada, 2006, 1529–1536.

[28] R.N. Mantegna, Fast, accurate algorithm for numerical simulation of Lévy stable stochastic processes, *Physical Review E*, *49*(5), 1994, 4677–4683.

Biographies

Baohua Qiang received his B.E. and M.A. degrees from Southwest University in 1996 and 2002, respectively, and the Ph.D. degree from Chongqing University in 2005. He moved to the University of Illinois as a visiting scholar from May to August in 2007. He had completed his postdoctoral research at the South China University of Technology from 2007 to 2009. Today he is a professor of Guilin University of Electronic Technology. His major research interests are web information processing, intelligent search, massive data processing, and network information integration.



Tian Zhao received his B.E. degree from Guilin University of Electronic Technology in 2015. Now he is a master degree candidate of Guilin University of Electronic Technology. His research interests are deep learning and information mining.

Wu Xie received his B.E. and M.A. degrees from Guilin University of Electronic Technology in 2002 and 2005, respectively, and the Ph.D. degree from Xidian University in 2015. Now, he is a senior experimenter of Guilin University of Electronic Technology. His major research interests are data mining, big data, cloud computing, and information processing.

Hong Zheng received her B.E. and M.A. degrees from Guilin University of Electronic Technology in 1997 and 2006, respectively, and the Ph.D. degree from Xiamen University in 2018. Now, she is a lecturer in Guilin University of Electronic Technology. Her major research interests are medical image reconstruction, image processing, and machine learning.

Haoning Sun received his B.E. degree from Industrial and Commercial College, Hebei University in 2014. Now, he is a master degree candidate of Guilin University of Electronic Technology. His research interests are data mining and machine learning.

Jinlong Chen received his B.E. degree from Guilin University of Electronic Technology in 2001. Now, he is a senior experimenter of Guilin University of Electronic Technology. His major research interests are information processing and software engineering.