

## Research Article

# An Improved Spectral Clustering Community Detection Algorithm Based on Probability Matrix

Shuxia Ren , Shubo Zhang, and Tao Wu

Department of Computer Science and Technology, Tianjin Polytechnic University, Tianjin 300387, China

Correspondence should be addressed to Shuxia Ren; [t\\_rsx@126.com](mailto:t_rsx@126.com)

Received 21 March 2020; Accepted 28 April 2020; Published 4 June 2020

Academic Editor: Jianquan Lu

Copyright © 2020 Shuxia Ren et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The similarity graphs of most spectral clustering algorithms carry lots of wrong community information. In this paper, we propose a probability matrix and a novel improved spectral clustering algorithm based on the probability matrix for community detection. First, the Markov chain is used to calculate the transition probability between nodes, and the probability matrix is constructed by the transition probability. Then, the similarity graph is constructed with the mean probability matrix. Finally, community detection is achieved by optimizing the NCut objective function. The proposed algorithm is compared with SC, WT, FG, FluidC, and SCRW on artificial networks and real networks. Experimental results show that the proposed algorithm can detect communities more accurately and has better clustering performance.

## 1. Introduction

With the development of information technology, the interactions among the complex systems of biology, sociology, and other fields are getting closer and closer. It is of great theoretical significance and practical value to obtain relevant information from real complex systems. According to graph theory, most real complex systems where their internal entities have rich associations can be abstracted into complex networks, such as neural networks, power networks, and social networks. In addition to the small-world and scale-free properties, complex networks have an extremely important community structure [1]. Community is a mesoscopic structure in which nodes from the same community are closely connected to each other, but nodes from different communities are sparsely connected. It is playing an important role in revealing the topological structure and functional features of complex networks. In recent years, community detection has been a popular research field for searching information, analysing function, and forecasting behaviour.

Community detection is a process of dividing a network into many clusters according to certain relationships among nodes. Moreover, community detection can classify nodes based on the topological structure of the network. It can

reveal the hidden hierarchical structure of the real network and improve the performance and efficiency of storing, processing, and analysing network data. So far, there are many methods for community detection, such as spectral bisection algorithm [2], graph segmentation algorithm [3], heuristic algorithm [4], and objective optimization algorithm [5].

## 2. Related Work

Community detection is an important branch of complex networks. Among the traditional community detection algorithms, the most famous algorithm is the spectral analysis algorithm based on network topology, which is referred to as spectral clustering [2] in the following. Its main idea is eigen-decomposing the similarity matrix of the network to obtain the main eigenvectors for finding communities. Not only is the spectral clustering algorithm applicable for a variety of data structures but also it utilizes dimensionality reduction to reduce computational complexity. Consequently, scholars began to research spectral clustering and optimize and expand on it.

Qin et al. [6] proposed a multisimilarity spectral method for clustering dynamic networks. It detects communities by

bootstrapping the clustering of different similarity measures. Ulzii and Sanggil [7] designed an agglomerative spectral clustering method with conductance and edge weights. The most similar nodes are agglomerated based on eigenvector space and edge weights. Ding et al. [8] explored the equivalence relation between the nonnegative matrix factorization and spectral clustering and developed a semi-supervised spectral clustering algorithm.

Spectral clustering typically constructs a similarity matrix with Euclidean distance between nodes. However, the Euclidean distance may lose the hidden relationship among nodes. As a result, the similarity matrices cannot contain complete community information. Clustering performance is not satisfied. If the constructed similarity matrix can approach the ideal matrix, the spectral clustering algorithm will have better clustering performance. Hence, constructing an excellent similarity matrix is the key to the spectral clustering community detection algorithm.

Nataliani and Yang [9] proposed a new affinity matrix generation method by using neighbour relation propagation principle. The method can increase the similarity of point pairs that should be in the same cluster. But the distance threshold is easily affected by outside points or noise points. Beauchemin [10] presented a method to build affinity matrices from a density estimator relying on K-means with subbagging procedure. However, this method would not work well when manifold proximity exists. Zhang and You [11] developed an approach based on a random walk to process the similarity matrix. The pairwise similarity is not only related to the two points but also related to their neighbours. However, the threshold of neighbouring nodes is set manually, and the stability of clustering is bad.

Although many community detection algorithms based on optimizing a similarity graph have been proposed, how to construct the similarity graph that can correctly reflect the community structure has not been solved. Consequently, this paper focuses on the transition probability between nodes to calculate the similarity, presents the concept of probability matrix, and proposes an improved spectral clustering community detection algorithm based on the probability matrix.

### 3. Improved Spectral Clustering Algorithm

**3.1. Constructing a Similarity Graph by Probability Matrix.** The similarity graph of spectral clustering is constructed by calculating the similarity between nodes. In this section, the similarity between nodes is calculated by the transition probability among nodes. And the related concepts of probability matrix and mean probability matrix are introduced. Then, the similarity graph is constructed based on the mean probability matrix.

**3.1.1. Transition Probability.** A Markov chain is a stochastic process of variables with Markov property, describing a sequence of states. The state changes over time, and the next state of the sequence depends on the current state [12]. The

possibility of transition between states is called the transition probability.

Given a network  $N$ , the number of nodes is  $n$ , the adjacency matrix of  $N$  is  $W$ . The probability that node  $i$  reaches to node  $j$  after one step is the 1st transition probability, which can be defined as

$$pr_{ij} = \frac{w_{ij}}{\sum_{j=0}^{n-1} w_{ij}}. \quad (1)$$

The 1st transition matrix  $Pr$  is a matrix composed of entry  $pr_{ij}$ , then  $Pr = D_W^{-1} \cdot W$ , where  $D_W = \text{diag}(d_{W0}, d_{W1}, \dots, d_{W_{n-1}})$ ,  $d_{Wi} = \sum_{j=0}^{n-1} w_{ij}$ .

The probability that node  $i$  reaches to node  $j$  after  $l$  steps is the  $l$ -th transition probability. And the matrix formed by  $l$ -th transition probability is called the  $l$ -th transition matrix  $Pr_l$ . According to the properties of the Markov chain, we can get

$$Pr_l = Pr^l. \quad (2)$$

$Pr^l$  denotes that  $Pr$  is multiplied by  $l$  times.

#### 3.1.2. Probability Matrix

**Definition 1.** Given a network  $N(V, E)$ , considering that the transition probability from node  $i$  to node  $j$  is  $p_{ij}$ , then the probability matrix of  $N$  is a  $V \times V$  matrix composed of  $p_{ij}$ . The probability matrix can be referred to as  $P$ ,  $P = (p_{ij})$ .

The probability matrix describes the transition probability between nodes in the network. The 1st transition probability can reflect the most direct relationship between the node and its adjacent nodes, but there is a lack of hidden relationship with the nonadjacent nodes. The multistep transition probability can include more neighbour nodes, reflecting the multiple complex associations among nodes. However, the multistep walk may fail to reach the adjacent nodes, which could weaken the relationship with the adjacency nodes. Consequently, we propose a method for constructing the probability matrix based on the accumulation of weighted multiorder transition matrices, and  $P$  can be defined as

$$P = \sum_{i=1}^t w_i Pr_i, \quad (3)$$

where  $i$  denotes that the current state of the Markov chain is at time  $i$ ,  $t$  refers to the size of the Markov chain, called time scale,  $Pr_i$  is the  $i$ -th transition matrix, and  $w_i$  represents the weight of  $Pr_i$ ,  $i \in [2, t]$ ,  $w_i \in [0, 1]$ ,  $\sum_{i=0}^t w_i = 1$ .

**3.1.3. Mean Probability Matrix.** The time scale is the key to calculating the similarity of nodes. But the optimal time scale of different networks is not necessarily the same. There will be mistakes in using a fixed time scale. In order to reduce the influence of parameters  $t$  and  $w$ , the mean probability matrix obtained from the mean values of  $P$  with different time scales is proposed.

*Definition 2.* Given a network  $N(V, E)$ , considering that its probability matrix is  $P$  and the time scale is  $t$ , then the mean probability matrix is a  $V \times V$  matrix composed of the average of  $P_1, P_2, \dots, P_t$ . According to equations (2) and (3), the mean probability matrix can be referred to as  $P_M$ :

$$P_M = \frac{\sum_{i=1}^t P_i}{t} = \frac{\sum_{i=1}^t \sum_{j=1}^i P_{ij}}{t}. \quad (4)$$

Not only does the time scale  $t$  of  $P_M$  provide the size of time scale for each  $P$ , but more importantly, it specifies the number of summing probability matrices. It could take different  $P$  to average the error caused by  $t$  and  $w$ . The mean would reduce mistakes, and the different value of  $t$  does not cause a great error. As a result, the value of  $t$  can be randomly chosen, but in order to reduce the computational complexity, we set  $t$  to be [5, 13].

The weight  $w_j$  also represents the weight of the  $j$ -th transition matrix. According to Definition 2, we can get that  $i$  gradually changes, and the number of corresponding weights  $w_j$  also gradually changes. To satisfy the constraint,  $w_j$  is defined as

$$w_j = \frac{ws_j}{\sum_{h=1}^t ws_h}, \quad (5)$$

where  $ws$  is a set of weights of size  $t$ , and it is artificially set, satisfying  $ws_1 > ws_2 \geq \dots \geq ws_t$ .

### 3.2. Improved Spectral Clustering Algorithm Based on Mean Probability Matrix

*3.2.1. Constructing the New Similarity Graph.* The similarity matrix  $W_p$  is constructed by the mean probability matrix  $P_M$ . Given a network  $N$ , the mean probability matrix of  $N$  is  $P_M$ ; then, the similarity between node  $i$  and  $j$  can be defined as

$$w_{P_{ij}} = \begin{cases} P_{M_{ij}} & \text{for } i \neq j, \\ 0 & \text{for } i = j, \end{cases} \quad (6)$$

where  $w_{P_{ij}}$  denotes the  $i, j$ -th entry of  $W_p$ , and  $P_{M_{ij}}$  refers to the  $i, j$ -th entry of  $P_M$ .

The similarity matrix of the traditional spectral clustering is a symmetric matrix, which is beneficial to calculate the Laplacian matrix  $L$ . Although  $W_p$  is not a symmetric matrix,  $W_p$  has special properties and can also construct  $L$ . The properties of  $W_p$  are as follows:

$L_W = D - W_p$ , where  $D$  is a diagonal matrix,  $D_{ii} = \sum_{j=1}^n w_{P_{ij}}$ , the entries on the diagonal are positive,  $W_p$  is a matrix with nonnegative entries, its diagonal entries are all 0, and each row of entries is not all 0. To sum up, it turns out that  $L_W$  is a matrix where all the diagonal elements are positive, and the other elements are negative. Then, we obtain that  $L_W$  is invertible.

For any vector  $f$ ,  $L_W$  can satisfy

$$\begin{aligned} f^T L_W f &= f^T D f - f^T W_p f = \sum_{i=1}^n d_i f_i^2 - \sum_{i,j=1}^n w_{P_{ij}} f_i f_j \\ &= \frac{1}{2} \left( \sum_{i=1}^n d_i f_i^2 - 2 \sum_{i,j=1}^n w_{P_{ij}} f_i f_j + \sum_{j=1}^n d_j f_j^2 \right) = \frac{1}{2} \sum_{i,j=1}^n w_{P_{ij}} (f_i - f_j)^2. \end{aligned} \quad (7)$$

As a result,  $L_W$  is a Laplacian matrix, and  $W_p$  can construct a similar graph of spectral clustering.

*3.2.2. NCut Objective Function.* Spectral clustering has many different objective functions. The purpose of the objective functions is to find a partition of the network such that the edges between different communities have lower weight and the edges within the same community have a higher weight. In other words, nodes in different clusters are dissimilar from each other, and nodes within the same cluster are similar to each other.

The more popular functions are RatioCut [13] and NCut [14]. RatioCut focuses on maximizing the number of nodes in the community, while NCut pays attention to maximizing the weights in the community. Given a network  $N(V, E)$ , they can be defined as

$$\text{RatioCut}(C_1, C_2, \dots, C_K) = \frac{1}{2} \sum_{k=1}^K \frac{W(C_k, \bar{C}_k)}{|C_k|}, \quad (8)$$

$$\text{NCut}(C_1, C_2, \dots, C_K) = \frac{1}{2} \sum_{k=1}^K \frac{W(C_k, \bar{C}_k)}{\text{vol}(C_k)},$$

where  $C_k$  denotes the set of nodes in the community  $k$ ,  $C_a \cap C_b = \emptyset$ ,  $C_1 \cup C_2 \dots \cup C_K = V$ ,  $a \neq b$ ,  $a, b \in \{1, 2, \dots, K\}$ ,  $K$  represents the number of communities,  $\bar{C}_k$  refers to the complement of  $C_k$ ,  $\bar{C}_k = V - C_k$ ,  $W(C_k, \bar{C}_k) = \sum_{i \in C_k, j \notin C_k} W_{P_{ij}}$ ,  $|C_k|$  is the number of nodes in  $C_k$ , and  $\text{vol}(C_k)$  is the sum of the weights of edges in  $C_k$ ,  $\text{vol}(C_k) = \sum_{i \in C_k} \sum_{j=1}^{|V|} w_{P_{ij}}$ .

The number of nodes in the community does not mean that the weight in the community is high. In comparison, NCut is more consistent with the clustering strategy of spectral clustering. Therefore, we choose NCut as the objective function of the proposed algorithm. Combined with equation (6), the objective function can be optimized as

$$\arg \min_F \text{tr}(F^T D^{-1/2} L_W D^{-1/2} F) \text{ s.t. } F^T F = I. \quad (9)$$

$F$  is a matrix composed of vectors  $f$ , and  $I$  is the identity matrix.  $F$  can be obtained by solving the first  $K$  smallest eigenvectors of  $D^{-1/2} \cdot L_W \cdot D^{-1/2}$ . However, a little information is missing due to dimension reduction, resulting in the fact that  $F$  cannot fully indicate the attributes of nodes. Therefore, taking a traditional clustering on  $F$ , such as  $K$ -means, can divide the network into  $K$  communities more accurately in the end.

Input network  $N$ , adjacency matrix  $W$ , community number  $K$ , time scale  $t$ , and a set of weights  $w$   
 Output  $K$  communities

- (1) Compute the 1st transition matrix  $P_r$  according to equation (1)
- (2) Compute the mean probability matrix  $P$  according to equation (4)
- (3) Construct the similarity matrix  $W_p$  according to equation (6)
- (4) Construct the unnormalised Laplacian matrix  $L_W$  according to the property 1 of  $W_p$  in Section 3.2.1
- (5) Construct the normalized Laplacian matrix  $L_n$  with  $L_n = D^{-1/2} \cdot L_W \cdot D^{-1/2}$
- (6) Compute the first  $K$  eigenvectors of  $L_n$ , referred to as  $U$
- (7) Consider the rows of  $U$  as nodes, and use K-means to cluster them into  $K$  communities

ALGORITHM 1: Improved spectral clustering algorithm based on the probability matrix.

TABLE 1: The hyperparameters of the LFR benchmark.

Parameter	Description	Value
$\beta$	Power-law index of the degree distribution	3
$\gamma$	Power-law index of community size	1.5
<i>ave_deg</i>	Average degree of each node	10
<i>min_com</i>	Minimum number of nodes in any community	30
<i>seed</i>	Seed number of random number generator	10

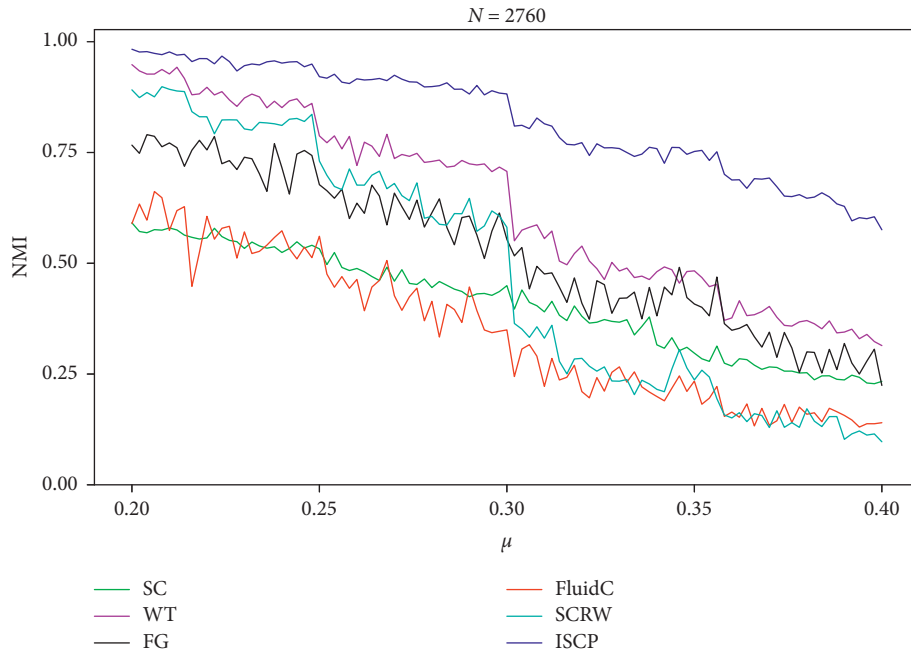


FIGURE 1: Performance in NMI of six algorithms with various  $\mu$ . In the experiments, the network size is 2760, the initial value of  $\mu$  is 0.2, the step length of  $\mu$  is 0.002, and the number of iterations is 101.

3.2.3. *The Main Steps of the Algorithm.* The main steps of the improved spectral clustering algorithm is given in Algorithm 1.

## 4. Experiments and Analyses

The experimental data includes artificially generated networks and real networks. On the one hand, we use the LFR benchmark network [15] to generate the networks and evaluate the quality of community detection by normalized

mutual information (NMI) [16]. On the other hand, we adopt several real networks and take the modularity ( $Q$ ) [17] as the evaluation index.

In order to show the performance of the improved spectral clustering algorithm (ISCP), ISCP is compared with SC [2], WT [18], FG [19], FluidC [20], and SCRW [11]. The experimental environment includes Intel 2.5 Hz i7-4710MQ CPU and 8 G RAM. The software platform is PyCharm 2018.1.2 (Community Edition) in Windows 10  $\times$  64.

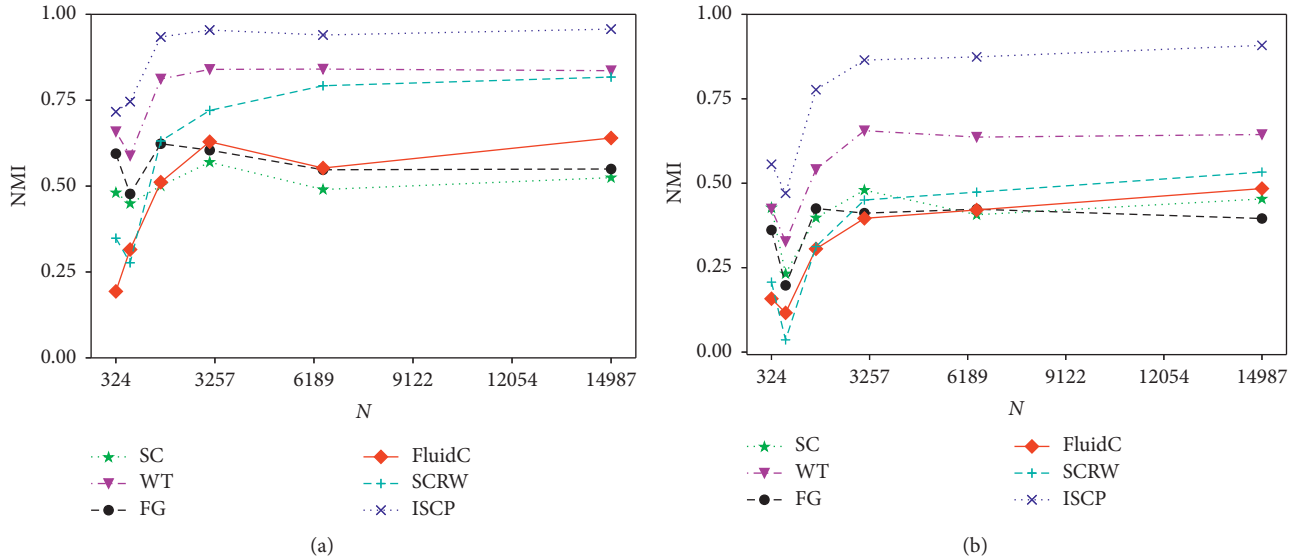


FIGURE 2: Performance in NMI of the six algorithms with different network sizes  $N$ . The network sizes are 324, 736, 1654, 3102, 6458, and 14987, respectively. In (a), the six algorithms are evaluated with  $\mu=0.3$ . Already in (b)  $\mu$  is set to 0.35.

4.1. *LFR Benchmark Networks.* The LFR benchmark networks are computer-generated networks, and they can produce different features of networks by adjusting some parameters. The experiments mainly use mixing parameter  $\mu$  ( $\mu$  denotes the average rate of edges connected with other communities,  $0 \leq \mu \leq 1$ ) and network size  $N$  to evaluate performance. To guarantee consistency, the detailed descriptions and values of other parameters are shown in Table 1.

Figure 1 shows the performance of the six algorithms on  $\mu$ . From Figure 1, we can get that the NMI trend of ISCP is smoother than other algorithms, and the NMI of ISCP is significantly higher than the other five algorithms. In [16], we can obtain that the larger the NMI is, the better the quality of community detection is. Overall, the clustering effect of ISCP is significantly better than the other five algorithms. In general, ISCP is more stable, and its convergence speed is faster.

Figure 2 demonstrates that the performance of the six algorithms on different network sizes  $N$ . As seen in Figure 2, the NMI of ISCP is higher than the other five algorithms. And as network size increases, its NMI increases. When the network size reaches 5000 or more, its NMI tends to be stable and stays around 0.9. Therefore, whether the order of magnitude of the network size is 1000 or 10,000, the clustering performance of ISCP is better than the other five algorithms.

4.2. *Real-World Networks.* The real-world networks have different topologies from the benchmark networks. To further evaluate the performance of the algorithms, 8 real-world networks are taken to do experiments. Moreover, it is necessary to normalize some real-world networks, such as eliminating self-loops and constructing a connected network. The detailed information of these networks is shown in Table 2.

TABLE 2: The hyperparameters of the LFR benchmark.

No.	Networks	Nodes	Edges
1	US airlines	332	2126
2	Erdos991	454	1421
3	Roget	1010	3656
4	Science	1461	3009
5	SCiNet	2729	10419
6	US power grid	4941	6594
7	Erdos992	6094	9949
8	PGP	10680	24316

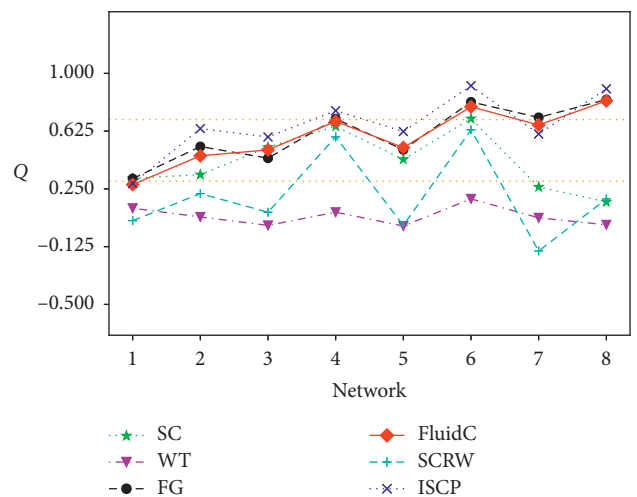


FIGURE 3: Performance in  $Q$  of the six algorithms with eight real-world networks. For each network,  $Q$  is the mean value obtained by clustering the same network for different community numbers  $K$ .  $K$  is from 20 to 100, the step length of  $K$  is 10, and the number of iterations is 9. (The two yellow reference lines are 0.3 and 0.7 respectively.)

The experiments take modularity  $Q$  to evaluate the clustering performance of the six algorithms. The range of  $Q$  is from  $-0.5$  to  $1$ . The larger  $Q$  is, the better the community detection performance will be.  $Q$  generally falls in about  $0.3$  to  $0.7$  in practice [17].

Figure 3 shows the performance of the six algorithms for clustering real-world networks. As shown in Figure 3,  $Q$  of ISCP is almost all above  $0.3$  and is larger than  $Q$  of the other algorithms. Although ISCP is not the best community detection algorithm for network 1 and network 7, its performance is very close to the best algorithm. Generally speaking, ISCP has excellent clustering performance and can cluster real-world networks more accurately.

## 5. Conclusions

Spectral clustering plays an important role in the field of community detection. It is an excellent community detection algorithm, but the traditional similarity graphs contain lots of incorrect information about the community structure. As a result, the performance of community detection is bad. Hence, this paper presents the probability matrix and proposes an improved spectral clustering community detection algorithm ISCP. A large number of experiments on benchmark networks and real-world networks show that ISCP is better than most traditional community detection algorithms and can more accurately cluster complex networks.

However, the ISCP will cost lots of time and space. Given a network  $N$ , the number of nodes is  $n$ , and the time scale is  $t$ . ISCP needs to multiply the transition probability matrix by  $t$  times to construct the similarity matrix. Even with the Fast Power algorithm, the time complexity of the algorithm will reach  $O(n^3lbt)$ . As the size of the network is larger, computing the similarity matrix will take more time and space. Moreover, ISCP is only applied to nonoverlapping complex networks. So the next step is to research how to optimize the computational complexity of the algorithm and how to cluster overlapping networks.

## Data Availability

The data cannot be released for the time being. When the relevant research is finished, we will release detailed research results.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## References

- [1] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.
- [2] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: analysis and an algorithm," *Advances in Neural Information Processing Systems*, vol. 14, pp. 849–856, 2002.
- [3] B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *Bell System Technical Journal*, vol. 49, no. 2, pp. 291–307, 1970.
- [4] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, 2008.
- [5] D. He, J. Liu, D. Liu, D. Jin, and Z. Jia, "Ant colony optimization for community detection in large-scale complex networks," in *Proceedings of the Seventh International Conference on Natural Computation*, pp. 1151–1155, Shanghai, China, July 2011.
- [6] X. Qin, W. Dai, P. Jiao, W. Wang, and N. Yuan, "A multi-similarity spectral clustering method for community detection in dynamic networks," *Scientific Reports*, vol. 6, no. 1, pp. 31454–31465, 2016.
- [7] N. Ulzii and K. Sanggil, "Social network community detection using agglomerative spectral clustering," *Complexity*, vol. 2017, Article ID 3719428, 10 pages, 2017.
- [8] S. Ding, H. Jia, M. Du, and Y. Xue, "A semi-supervised approximate spectral clustering algorithm based on HMRP model," *Information Sciences*, vol. 429, pp. 215–228, 2018.
- [9] Y. Nataliani and M. S. Yang, "Powered Gaussian kernel spectral clustering," *Neural Computing and Applications*, vol. 31, no. 1, pp. 557–572, 2019.
- [10] M. Beauchemin, "A density-based similarity matrix construction for spectral clustering," *Neurocomputing*, vol. 151, pp. 835–844, 2015.
- [11] X. Zhang and Q. You, "An improved spectral clustering algorithm based on random walk," *Frontiers of Computer Science in China*, vol. 5, no. 3, pp. 268–278, 2011.
- [12] J. Rhodes and A. Schilling, "Unified theory for finite Markov chains," *Advances in Mathematics*, vol. 347, pp. 739–779, 2019.
- [13] L. Hagen and A. B. Kahng, "New spectral methods for ratio cut partitioning and clustering," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 11, no. 9, pp. 1074–1085, 1992.
- [14] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.
- [15] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Physical Review E*, vol. 78, no. 4, pp. 46110–46115, 2008.
- [16] P. Zhang, "Evaluating accuracy of community detection using the relative normalized mutual information," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2015, no. 11, pp. 11006–11013, 2015.
- [17] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.
- [18] P. Pons and M. Latapy, "Computing communities in large networks using random walks," *Journal of Graph Algorithms and Applications*, vol. 10, no. 2, pp. 191–218, 2006.
- [19] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Physical Review E*, vol. 69, no. 6, pp. 133–138, 2004.
- [20] F. Parés, D. Garcia Gasulla, A. Vilalta et al., "Fluid communities: a competitive, scalable and diverse community detection algorithm," in *Proceedings of the Complex Networks & Their Applications*, pp. 229–240, Cham, Switzerland, November 2017.