# Comparison of multiple different overlapping community discovery algorithms

Weihua Li

College of Information Engineering,
Longyan University,
Tongxin Building, No. 1,
Dongxiao North Road, Xinluo District,
Longyan, Fujian 364012, China
Email: weihlwhua@yeah.net

**Abstract:** When analysing a real-world problem, it can be transformed into a complex network, and then its structure is divided to find out the overlapping part and study the hidden laws. In this paper, three overlapping community recognition algorithms are briefly introduced, i.e., overlapping community partitioning based on label propagation (OCPLP), footpad skin optical clearing agent (FSOCA) and the genetic algorithm (GA), and then the three artificial networks generated by the LFR tool are simulated and analysed through matrix laboratory (MATLAB) software. The results show that the increase of network complexity will reduce the recognition performance of the three algorithms. In the three algorithms, GA is relatively less affected, which always has the highest recognition performance. OCPLP is the most affected, which has the worst performance. In terms of the speed of recognition, GA is also the fastest, and OCPLP is the slowest. In summary, GA is more suitable for the search of overlapping communities in the network than that of OCPLP and FSOCA.

**Keywords:** overlapping community; overlapping community partitioning based on label propagation; OCPLP; footpad skin optical clearing agent; FSOCA; genetic algorithm.

**Reference** to this paper should be made as follows: Li, W. (2020) 'Comparison of multiple different overlapping community discovery algorithms', *Int. J. Web Based Communities*, Vol. 16, No. 1, pp.109–119.

**Biographical notes:** Weihua Li has a Masters in Engineering. Currently, he works in the School of Information Engineering, Longyan University. He is an experimenter. He is interested in computer application technology, software engineering technology and data mining technology.

## 1 Introduction

In reality, there are many complicated problems, and the same or similar problems can be better solved to promote the development of society by studying the laws (Cheng et al., 2018). However, the complexity of the real problems make the surface laws that can be visually discovered not very relevant and have no fundamental influence on the solution of problems. In order to dig hidden information, complex problems are transformed into a social network. The social network consists of multiple nodes and line segments between

nodes, in which a node represents an individual participating in it, and a line segment represents a connection between individuals. If individuals are 'interested by interest', the connection between them will be dense. If there is 'no speculation', the connection between them will be sparse or even disconnected. Finally, the social network presents a sparse and dense topology map (Sheikholeslami and Giannakis, 2018). Nodes that are densely connected to each other constitute a community. In real life, nodes cannot completely belong to only one community, and some nodes are connected with nodes in another community, causing overlap between two communities. The overlapping parts of the network often reflect the key information. At the same time, in the actual application, based on the characteristics of that the nodes in the overlapping community (Gui et al., 2018) are linked to multiple communities, local information can be disseminated to a larger extent. For example, in areas used for communication such as forums, different users (nodes) have different section tendencies. As a whole, a small number of nodes will have the same interest tendencies, but also participate in other interest-oriented communities. Therefore, when those nodes spread rumours, they will spread quickly to the whole Internet. Overlapping community identification algorithm can effectively find overlapping nodes and manage them pertinently to improve the efficiency of prevention and management. However, in reality, tens of thousands of nodes and segments constitute social networks, so it is very important to quickly identify algorithms of overlapping networks. Zhou (2015) proposed a hierarchical gamma process infinite edge model for the network without weight and vector, which could not only find overlapping communities, but also predict the missing edges to a certain extent. The simulation results proved the scalability and advanced performance of the model. Yu et al. (2015) proposed a community detection clustering algorithm based on the link-field-topic (LFT) model to detect overlapping groups of semantic communities. The simulation results verified the validity and feasibility of the LFT model. Wang et al. (2014) proposed a multi-attribute edge-centric collaborative clustering framework based on the user's check-in records and group attributes in social groups, which was used to discover overlapping hierarchical communities of social network users. The simulation results verified the effectiveness of the method. In this paper, three overlapping community recognition algorithms are briefly introduced, i.e., overlapping community partitioning based on label propagation (OCPLP), footpad skin optical clearing agent (FSOCA) and genetic algorithm (GA), and then the three artificial networks generated by the LFR tool are simulated and analysed through matrix laboratory (MATLAB) software.

## 2 The discovery algorithm based on OCPLP

OCPLP is based on tag propagation (Liu et al., 2015), i.e., the community of the corresponding node is divided by the attributes of the tag. The entire network abstraction is a network diagram which consists of multiple nodes and line segments, $G$. As shown in Figure 1, a circle represents a node, and a line segment indicates a link between the nodes, and the letter number in the node is its ID and is used as tag information in the community. The specific steps are as follows.

1 Random initialisation: a buffer, $b_i$ is set in each node before the OCPLP algorithm initialises the graph of network community node to store the change of the node labels, and the size is set as $N_b$. The subsequent initialisation is that each node randomly stores the ID of the adjacent node into its own buffer.

2 Label propagation: calculation is performed first according to formula (Li et al., 2016):

$$\begin{cases} p_i = \dfrac{n_i}{N_b n} \\ N_{ji}^e = p_i n_j^n N_b \\ s_{ji} = N_{ji}^a - N_{ji}^e \end{cases} \tag{1}$$

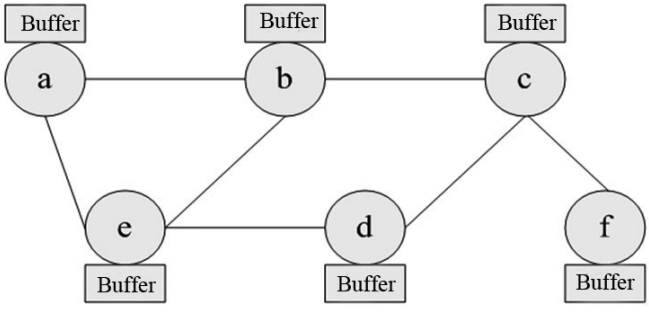
where, $p_i$ represents the global probability distribution of the label, $n_i$ represents the number of times that the $i$-th tag appears in all network node buffers, $n$ represents the total number of nodes, $N_{ji}^e$ represents the expected number of the $i$-th tag in the adjacent buffer of the j-th node, $n_j^n$ represents the number of neighbouring nodes of the $j$-th node, $s_{ji}$ represents the specificity of the $i$-th label in the $j$-th node buffer, and $N_{ji}^a$ represents the actual number of the $i$-th tag in the adjacent buffer of the $j$-th node. After $s_{ji}$ is obtained by calculation, the $i$-th tag corresponding to the largest $s_{ji}$ in the $j$-th node is added to the end of the buffer, and the first tag in the buffer is deleted. The above operations are repeated until the buffers of all nodes in the network no longer change.

3 Extraction of the division result: After the network node buffer is converged, the label with the largest number of nodes in the node buffer is set as the ID of the community to which the node belongs. When other nodes count the buffers of the respective nodes to get the ID of the node, the nodes with the same ID are grouped into one community, and the corresponding nodes are used as elements in the community.

4 1~3 are repeated $N$ times, $N$ kinds of results of community division are obtained, and the adjustment ranking indicator is used to select the best division result.

5 In the optimal division result, there will be a phenomenon in which a node belongs to two or more communities at the same time, and then the node belongs to the community overlap. When the coincident part is greater than the set threshold, the community is merged.

**Figure 1** Random initial map of network community node



## 3 The discovery algorithm based on FSOCA

FSOCA (Yuan et al., 2017) belongs to evolutionary algorithms, which also abstracts the network into the network diagram of Figure 1. The algorithm discovers overlapping communities by calculating the degree of connectivity between the nodes and the community. The steps of the algorithm are as follows.

1 Initial stage: the nodes in the network and their adjacent nodes form a community, the number of adjacent nodes is at least $k$, and the initial communities can overlap, and the nodes are allowed to stay in multiple high-connected communities or to leave. The community structure in the initial network is denoted as $C_0$.

2 Departure stage: After allocating the community in the initial stage, the distribution score of each node in the respective community is calculated, and the formula (Shan et al., 2016) is

$$\begin{cases} S_j^i = \begin{cases} \dfrac{\left|N_i\left(v_j\right)\right| - K + 1}{\left|C_i - K\right|} & \left|N_i\left(v_j\right)\right| > K \\ 0 \quad other \end{cases} \\ CS_j^i = \dfrac{\left|N_i\left(v_j\right)\right|}{\left|N\left(v_j\right)\right|} \end{cases} \tag{2}$$

where, $S_j^i$ represents the community connectivity score of node $j$ in the community $C_i$, $CS_j^i$ represents the connected score of the neighbouring node of node $j$, which indicates that the ratio of the neighbouring point in $C_i$, $N_i(v_j)$ represents the set of neighbouring nodes of node $j$ in $C_i$, and $N_i(v_j)$ represents the set of neighbouring nodes of node $j$. Then the departure of the peripheral nodes in the community is determined according to the community connectivity score. When the community connectivity score of the peripheral node is less than the set threshold, the peripheral node leaves the community, and the threshold is obtained by calculating the community connectivity score of node j in all communities. When the node in the community is less than $K$, the community is cleared. The above operation is repeated until there are no nodes to leave.

3 Expansion phase: after the departure phase is completed, when a node in the set of peripheral nodes of community is not joined to the community, and the neighbouring connectivity score of the node is greater than the set threshold, the node is added to the community $C_i$. After the expansion is completed, the next departure phase is entered, the departure and expansion phases are repeated until there are no peripheral nodes in a certain phase, and the iteration is stopped.

4 Deletion of duplicate community: in the process of calculation iteration of FSOCA, any initial community can join the new node, and the joined nodes will be not be checked whether they also belong to other communities, so there will be duplicate communities in the process of calculation. The formula which is to judge the community repeatability (Chakraborty et al., 2016) is:

$$sim\left(C_i, C_j\right) = \frac{\left|C_i \cap C_j\right|}{\min\left(\left|C_i\right|, \left|C_j\right|\right)} \tag{3}$$

The deletion operation is performed after the expansion phase and before the next departure phase. When the community repeatability exceeds the set threshold (OVL), one of the communities is deleted. When the threshold is set as 0, the community will be deleted as long as it has the same point, otherwise the actual situation is not matched, and the operation has no meaning. When it is set as 1, the deletion operation is performed only when it is exactly the same, otherwise it does not make sense. Therefore the threshold is generally set as a value in [0.5, 1).

## 4 The discovery algorithm based on GA

GA (Bello-Orgaz et al., 2018) is a simulation of the survival of the fittest in the biological world. The node locations and the connection types are taken as chromosomal genes coding, then the way of inheritance and variation in the natural world is imitated to iterate and screen, and finally the network community is identified. The steps are as follows.

1 Population initialisation: firstly, according to the total number of nodes in the network and the estimated number of communities, the average density of the community is obtained as the largest size of the community. Then initialisation is started. The first-level neighbour initialisation is performed first (the first-level neighbour refers to another node which is directly connected to the node). A node from the outside initial community is randomly selected, which is subsumed to the initial community along with its first-level neighbours. If the maximum size is reached, the initialisation is completed, otherwise the secondary neighbour initialisation is performed (secondary neighbour refers to neighbouring nodes that is indirectly connected to the node). A node from the community after the first-level neighbour initialisation is randomly selected, which is subsumed to the initial community along with its secondary neighbours, until the maximum size is reached. In the evolution of the population, there are not only crossovers and mutations, but also the strategy of migrating to the most attractive targets, i.e., when the nodes which are directly connected to node j are distributed in different communities, node *j* migrates to communities with more directly connected nodes.

2 Computational of fitness: In the iteration process of GA, the fitness is needed to guide the evolution of the population. In this study, a modular function that can quantitatively describe the community structure is used as the fitness function. The formula (Qi and Xie, 2017) is:

$$Q = \begin{cases} \dfrac{\sum\left(a_{ij} - \dfrac{K_i K_j}{2M}\right)}{2M} & \delta(i,j) = 1 \\ 0 \quad \delta(i,j) = 0 \end{cases} \tag{4}$$

where, $Q$ represents the modular function, the greater the modular function is, the stronger the structure is, $M$ represents the total number of edges between nodes in the

network, $i, j$ respectively represent any two nodes in the network, $K_i$, $K_j$ respectively represent the degree of two nodes, $a_{ij}$ represents an element of the $i$-th row and $j$-th column in the adjacency matrix, and $\delta(i, j)$ represents the relationship between two nodes. When the value of $\delta(i, j)$ is 1, the two points are in the same community, and when the value of $\delta(i, j)$ is 0, the two points are in different communities.

3 Genetic iteration: Genetic manipulation is the core of GA, including crossover and mutation. The first is the crossover operation. In the overlapping community discovery algorithm of this paper, the crossover operation of GA is achieved through the node exchange between the community boundaries. The node with the optimal external relevance on the community boundary is selected and is exchanged with a node on another community boundary. The probability of crossover is the probability of crossover operation performed between two community boundaries.

   The mutation operation is to select the appropriate community; a node is randomly selected from the community and transformed into a node in another community with the highest relevance to the selected community. The probability of mutation is the probability of mutation operation performed between the communities.

4 Termination of iteration: the genetic iterative operation is repeated until the fitness is stable or the iteration reaches the maximum set number, and the calculation result is output.

## 5 Simulation analysis

### *5.1 Experimental environment*

In this study, the experiment is carried out on the laboratory server, the operating system is Windows7, the CPU is Core I7, and the size of memory is 16 GB. The above three algorithms are simulated and analysed by MATLAB software (Zhou et al., 2018).

### *5.2 Experimental data*

In this study, three different structures of artificial overlapping network are generated by using LFR tool. As shown in Table 1, the artificial network variable of LFR1 is the probability of connection between the node and the external community, the variable of LFR2 is the number of overlapping nodes connected to the community, and the variable of LFR3 is the number of overlapping nodes.

**Table 1** Data set of artificial overlapping network

| *Type of artificial network* | *LFR1* | *LFR2* | *LFR3* |
|---|---|---|---|
| Total number of nodes | 10000 | 10000 | 10000 |
| Node average | 20 | 20 | 20 |
| Maximum node | 50 | 50 | 50 |
| Minimum number of community nodes | 50 | 50 | 50 |

**Table 1** Data set of artificial overlapping network (continued)

| *Type of artificial network* | *LFR1* | *LFR2* | *LFR3* |
|---|---|---|---|
| Maximum number of community nodes/number | 100 | 100 | 100 |
| Number of overlapping nodes | 1000 | 1000 | 1000~8000 |
| Number of overlapping nodes connected to the community | 5 | 2~8 | 5 |
| Probability of connection between nodes and external community | 0.1~0.8 | 0.2 | 0.2 |

### *5.3 Evaluation indicators*

In this study, normalised mutual information (NMI) is used to evaluate the performance of three overlapping community discovery algorithms. The calculation formula (Matta et al., 2018) is:

$$\begin{cases} NMI = \dfrac{-2\sum\limits_{ij} N_{ij} \log_2\left(\dfrac{N_{ij}N_l}{N_{i.}N_{.j}}\right)}{\sum\limits_{i} N_i \log_2\left(\dfrac{N_{i.}}{N_l}\right) + \sum\limits_{j} N_j \log_2\left(\dfrac{N_{.j}}{N_l}\right)} \\ N_l = \sum\limits_{ij} N_{ij} \end{cases} \tag{5}$$

where, $N_{ij}$ represents the public nodes between the actual community $C_i$ and identification communities $C_j$, $N_l$ represents the public nodes of all actual communities and identification communities, and $N_i$, $N_j$ respectively represent the sum of the $i$-th row and the $j$-th column in the confusion matrix.

### *5.4 Experimental parameters*

In OCPLP, the size of buffer is set as 5, and the threshold is set as 0.5. In FSOCA, the threshold OVL is set as 0.6. In GA, the size of initial population is set as 150, the maximum number of iterations is 250, the crossover probability is 0.7, and the mutation probability is 0.3.

### *5.5 Experimental results*

As shown in Figure 2 (mu represents the probability of connection between the nodes and external community), the complexity of LFR1 artificial network is determined by mu. It can be seen from Figure 2 that with the increase of mu, NMI of the three discovery algorithms all reduces. The reason is that the increase of mu enhances the communication between nodes in different communities, leading to more varied and complex network structure and finally making the three algorithms difficult to identify the correct network. At the same time, comparing the performance of the three algorithms, it can be seen that the performance of GA is always higher than that of the other two algorithms, and OCPLP has the worst performance and fastest drop, although the performance of the three algorithms is degraded because of the complex structure.

**Figure 2** Identification performance of the three algorithms in LFR1 network (see online version for colours)

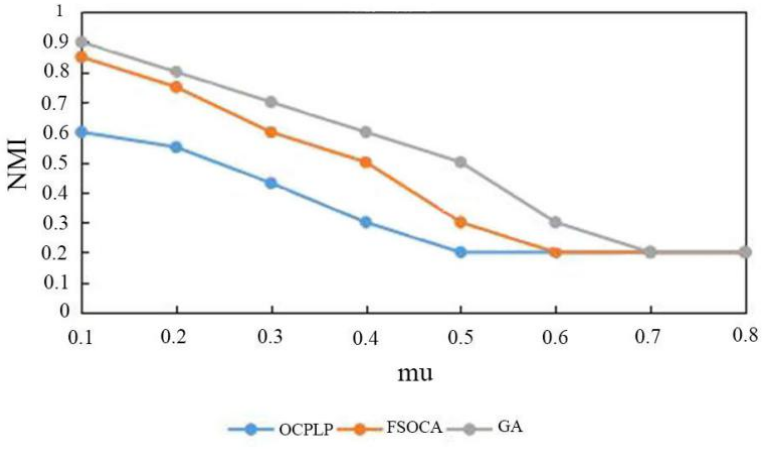
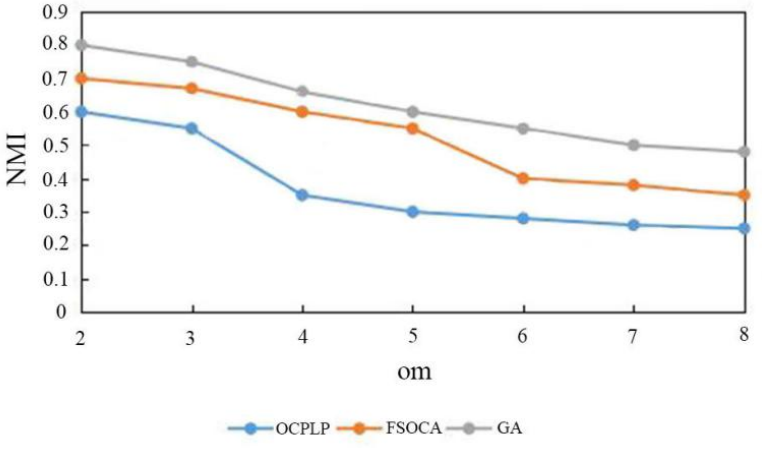



**Figure 3** Identification performance of the three algorithms in LFR2 network (see online version for colours)



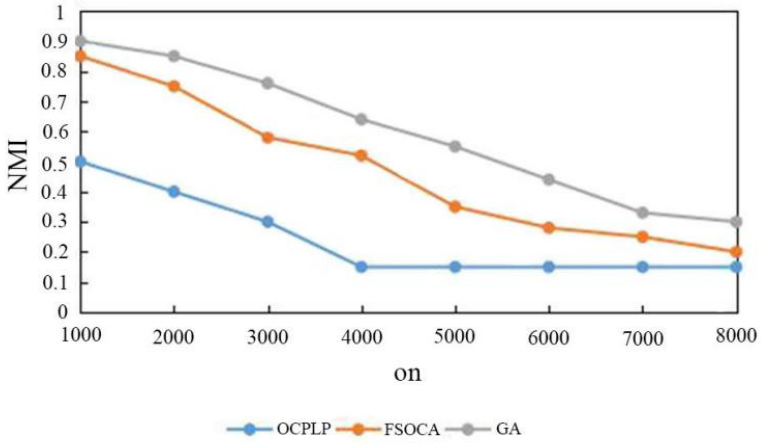

As shown in Figure 3 (om indicates the number of overlapping nodes connected to the community), and the complexity of the LFR2 artificial network is determined by om. It can be seen from Figure 3 that NMI of the three algorithms decreases as the number of overlapping nodes connected to the community increases, that is, the recognition performance is degraded. However, in this process, the recognition performance of GA is always higher than that of the other two algorithms, the performance of OCPLP is the worst, and GA is slower than the other algorithms in terms of the decrease speed with the increase of om.

As shown in Figure 4 (on indicates the number of overlapping nodes), the complexity of the LFR3 artificial network is determined by on. It can be seen from Figure 4, as the number of overlapping nodes increases, NMI of the three algorithms decreases, that is, the recognition performance is degraded. However, in this process, the recognition

performance of GA is always higher than the other two algorithms, and the performance of OCPLP is the worst.

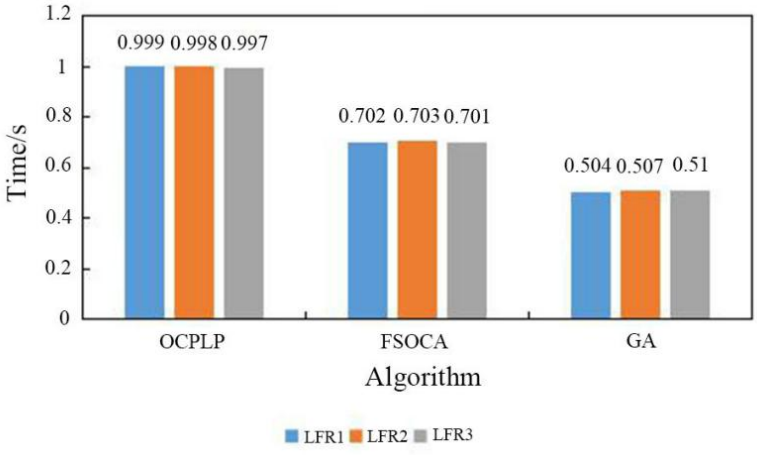
In summary, the probability of the connection between the node and the external community, the number of overlapping nodes connected to the community and the number of overlapping nodes all will increase the overlapping part of the community in the network and the connection between the nodes, resulting in blurred boundaries between the communities, thus the recognition performance of the three algorithms is inevitably reduced. In the three algorithms, the performance of OCPLP and FSOCA largely depends on the set threshold and the degree of connection between the communities, while GA uses the way of crossover and mutation to adjust the attribution of nodes. The increase of the edges between the communities is beneficial to genetic operations to a certain extent, so GA is relatively less affected by the complexity of network.

**Figure 4** Identification performance of the three algorithms in LFR3 network (see online version for colours)




The average time consumption of the three algorithms in the three artificial networks is counted. As shown in Figure 5, the average time consumed by OCPLP in LFR1, LFR2 and LFR3 is 0.999 s, 0.998 s and 0.997 s respectively; the average time consumed by FSOCA in LFR1, LFR2 and LFR3 is 0.702 s, 0.703 s and 0.701 s respectively; the average time consumed by GA in LFR1, LFR2 and LFR3 is 0.504 s, 0.507 s and 0.510 s respectively. It can be seen from Figure 5 that the time consumed by GA is the least in the artificial network of the same complexity, and the time consumed by OCPLP is the most. It can be found from Figure 5 that the average time consumed by the same algorithm in the three networks is similar. The reason is that the parameters in the network, such as the total number of nodes, node average, node maximum and maximum, minimum number of community nodes, etc. are the same, although the mu, om, and on are different in the three networks. Taking the average of time consumption is equivalent to take the average of the parameters of the network. After the average, the parameters of the three networks are basically the same, so the average time consumption of the same algorithm in the three networks is not much different.

**Figure 5** Calculation time of the three algorithms (see online version for colours)




## 6 Conclusions

In this paper, three overlapping community recognition algorithms are briefly introduced, i.e., OCPLP, FSOCA and GA, and then the three artificial networks generated by the LFR tool are simulated and analysed through MATLAB software. The results are as follows.

1 With the increase of the complexity of network structure, the recognition performance of the three algorithms for the network overlapping community is degraded. In this process, the recognition performance of GA is always the best, and it is relatively less affected, while OCPLP has the worst recognition performance and fastest drop.

2 When the three algorithms identify network overlapping communities, the recognition speed of GA is highest, and the recognition speed of OCPLP is lowest.

Although the performance of the above three overlapping community recognition algorithms is different, they can identify overlapping communities in the whole network. In practical applications, for example, in e-commerce marketing, a small number of people with common interests or shopping desires can be found by overlapping community algorithm, and these people have different communities. Therefore, the maximum effect can be achieved and the marketing cost can be reduced by implementing marketing strategies on those people.

## References

Bello-Orgaz, G., Salcedo-Sanz, S. and Camacho, D.A. (2018) 'Multi-objective genetic algorithm for overlapping community detection based on edge encoding', *Information Sciences*, Vol. 462, pp.290–314.

Chakraborty, T., Kumar, S., Ganguly, N., Mukherjee, A. and Bhowmick, S. (2016) 'GenPerm: a unified method for detecting non-overlapping and overlapping communities', *IEEE Transactions on Knowledge & Data Engineering*, Vol. 28, No. 8, pp.210–2114.

Cheng, J., Wu, X., Zhou, M., Gao, S., Huang, Z. and Liu, C. (2018) 'A novel method for detecting new overlapping community in complex evolving networks', *IEEE Transactions on Systems Man & Cybernetics Systems*, Vol. PP(99), pp.1–13.

Gui, C., Zhang, R., Hu, R., Huang, G. and Wei, J. (2018) 'Overlapping communities detection based on spectral analysis of line graphs', *Physica A Statistical Mechanics & Its Applications*, Vol. 498, No. 4, pp.50–65.

Li, C.Y., Tang, Y., Lin, H., Yuan, C.Z. and Mai, H.Q. (2016) 'Parallel overlapping community detection algorithm in complex networks based on label propagation', *Scientia Sinica*, Vol. 46, No. 2, p.212.

Liu, K., Huang, J., Sun, H., Wan, M., Qi, Y. and Li, H. (2015) 'Label propagation based evolutionary clustering for detecting overlapping and non-overlapping communities in dynamic networks', *Knowledge-Based Systems*, Vol. 89, No. C, pp.487–496.

Matta, J., Obafemiajayi, T., Borwey, J., Sinha, K., Wunsch, D. and Ercal, G. (2018) 'Node-based resilience measure clustering with applications to noisy and overlapping communities in complex networks', *Applied Sciences*, Vol. 8, No. 8.

Qi, W. and Xie, J. (2017) 'A two-dimensional genetic algorithm for identifying overlapping communities in dynamic networks', *IEEE International Conference on Tools with Artificial Intelligence*.

Shan, J., Shen, D.R., Nie, T.Z., Kou, Y. and Yu, G. (2016) 'Searching overlapping communities for group query', *World Wide Web*, Vol. 19, No. 6, pp.1179–1202.

Sheikholeslami, F. and Giannakis, G.B. (2018) 'Identification of overlapping communities via constrained egonet tensor decomposition', *IEEE Transactions on Signal Processing*, Vol. 66, No. 21, pp.5730–5745.

Wang, Z., Zhang, D., Zhou, X.S., Yang, D.Q. and Yu, Z.W. (2014) 'Discovering and profiling overlapping communities in location-based social networks', *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 44, No. 4, pp.499–509.

Yu, X., Yang, J. and Xie, Z.Q. (2015) 'A semantic overlapping community detection algorithm based on field sampling', *Expert Systems with Applications*, Vol. 42, No. 1, pp.366–375.

Yuan, P., Wei, W. and Song, M. (2017) 'Ties in overlapping community structures: strong or weak?', *IEEE Access*, Vol. 5, No. 99, pp.10012–10016.

Zhou, M. (2015) 'Infinite edge partition models for overlapping community detection and link prediction', *Computer Science*, pp.1135–1143.

Zhou, X., Zhao, X.H., Liu, Y.H. and Sun, G. (2018) 'A game theoretic algorithm to detect overlapping community structure in networks', *Physics Letters A*, Vol. 382, No. 13, pp.872–879.