Decision Science Letters 9 (2020) 547–558

Contents lists available at GrowingScience

Decision Science Letters

homepage: www.GrowingScience.com/dsl

An overlapping community detection algorithm based on local community and information flow expansion (LCFE) in weighted directed networks

Erfan Mohebiju[a] and Mehdi Ghazanfari[b*]

[a]*MSc student in industrial engineering, Iran University of Science and Technology, Tehran, Iran*
[b]*Professor of industrial Engineering, Iran University of Science and technology, Tehran, Iran*

C H R O N I C L E

Article history:
Received July 14, 2020
Received in revised format:
July 25, 2020
Accepted August 10 2020
Available online
August 10, 2020

Keywords:
Social Network Analysis
Community detection
Modularity
Similarity Index

A B S T R A C T

Community detection has gained much attention during the past few decades. So many algorithms have been developed to tackle this problem. In previous related works the weight of the edges and directionality were not considered at the same time in the models. Considering weights and directionality makes the models more realistic and prevents the loss of information in the network. In this article, we propose an overlapping community detection algorithm for networks with weighted and directed edges. We used the concept of information flows among the vertices i.e. the more flows exist in a community, the stronger the community. We implemented the concept of flow using weighted closed flows starting from a given node and ending to the same node. By using the mentioned assumption we developed a new modularity measure called weighted flow modularity (WFM) based on M function modularity. In addition, we developed an overlapping score criteria which considers overlap in vertices and edges at the same time and is much faster in the terms of run time. We compared the community detection results in terms of accuracy and running time with Order statistics local optimization method (OSLOM) on 74 LFR benchmark networks using normalized mutual information score. We also implemented the community detection process using LCFE on real world datasets and evaluated the community detection results using EQ measure. The experimental analysis results show that the LCFE is more accurate in most cases and is competitive in other cases with OSLOM.

1. Introduction

By development of information technology and trilling vast amounts of data in its beds the importance of social networks has been felt more than ever. Biological networks such as protein-protein interactions, online social networks, collaboration networks such as author's citation networks are examples of social networks (Girvan & Newman, 2002). Social network analysis reveals fundamental and strong insights about the modern world. That's why the amount of research conducted in the field of network science has been increased. One can simply describe a network as a set of nodes and edges depicting the interactions between them (Badiee & Ghazanfari, 2018). As an established fact in science of networks, the effect of the structure on the system is inescapable (Kermani, Badiee, Aliahmadi, Ghazanfari, & Kalantari, 2016). Some nodes in the network show a particular similarity to others. These nodes are divided into groups which are densely connected to each other while have sparse connections with other components of the network. The procedure in which one can find these groups is called community detection or clustering or graph partitioning and these groups are called clusters, partitions or communities. But still there is not a clear definition on the graph clustering problem both in undirected and directed graphs (Malliaros & Vazirgiannis, 2013). In this paper we will introduce an overlapping community detection algorithm for weighted and directed networks. The algorithm starts by calculating betweenness for every edge. Then based on the higher betweenness values the ends of the edge are selected as a local community, and the local communities will be expanded

* Corresponding author. Tel.: +982177240000
E-mail address: mehdi@iust.ac.ir (M. Ghazanfari)

doi: 10.5267/j.dsl.2020.8.003

through optimizing a modularity function called WFM. In the next step an overlapping score based on the similarity concept is introduced and communities with higher overlapping score will be merged. At the end, the homeless nodes are added to communities based on the value of their fitness. After the last phase, if a node is still homeless, we divide it into outlier group.

This paper is organized as following: In the second part we have a literature review on community detection algorithms in weighted directed networks. In the third section we present the model and the algorithm. In the fourth section we have numerical example and in the fifth section we have a conclusion and suggestions for future works respectively.

2. Literature review

Dealing with edge directionality have been an issue for a long time and various methods have been proposed to tackle this problem. In many works for convenience the directed graph is transformed into the undirected version thus all the algorithms for undirected community detections can be applied. This approach is called naïve graph transformation (Malliaros & Vazirgiannis, 2013). This method causes a huge loss in data and many vital information could be ignored. In some works the graphs have been transformed to undirected ones but the directionality is somehow maintained. There are two different approaches to deal with directionality in this context. The first approach transforms (converts the symmetric adjacency matrix to an asymmetric one), the directionality to edge weight and keeps the graph as unipartite. For example a directed network can be symmetrized through a two stage process (Satuluri & Parthasarathy, 2011). The idea behind the two stage transformation originates from a fact that a clustering algorithm should not solely depend on the density of the nodes but also the similarity in incoming and outgoing edges should be taken into consideration. In the first step multiple ways for symmetrizing an asymmetric adjacency matrix is proposed and in the second step an ordinary community detection approach can be used. A network can be symmetrized based on its embedding which can be considered indirectly as a transformation to an undirected weighted network; Laplacian matrix can be considered as an embedding (Lai, Lu, & Nardini, 2010b). Edge directionality can be extracted using a PageRank random walk and replace the directionality with edge weights (Lai, Lu, & Nardini, 2010a). The community detection process can be started using core nodes in the network and then, expanding the core nodes in their neighborhood to extract final community structure of the network (Long & Li, 2017).

In the second category the directionality is modeled through converting the graph to a bipartite network. In some works a scheme is used to transform the directed graph to an undirected bipartite graph ((Guimerà, Sales-Pardo, & Amaral, 2007; D. Zhou, Schölkopf, & Hofmann, 2005). In some other works an objective function is developed to tackle the directionality in graph clustering. In previous works the nature of the directed network was changed but in these methods, objective functions are developed to directly deal with the clustering problem of directed networks. In the first category a modularity function is developed for directed networks. Modularity is a criteria for assessing the quality of clusters (Newman & Girvan, 2004). The modularity function is generalized for directed networks based on reducing the initial size of the network while keeping the modularity intact (Arenas, Duch, Fernández, & Gómez, 2007). A modularity function for directed networks based on the original modularity was introduced and supposed that the modularity can be expressed through eigenvalues and eigenvectors of a specific matrix called modularity matrix (Leicht & Newman, 2008). LinkRank algorithm emphasizes on the edges other than nodes in the process of community detection (Kim, Son, & Jeong, 2010). The scalable Louvain algorithm with maximizing the modularity was extended and thus a brand new community detection algorithm was developed for directed networks (Dugué & Perez, 2015). Regularize asymmetric non-negative matrix factorization (RANMF) was developed which is based on an objective function with pairwise comparison of nodes (Tosyali, Kim, Choi, & Jeong, 2019). a consensus clustering algorithm for directed networks called ConClus which is comprised of three sub algorithms was developed. The algorithm mostly relies on a fitness function and a neural network providing intervals as resolution parameters (Santos, Carvalho, & Nascimento, 2016). An overlapping community detection algorithm for directed networks based on edge betweenness modularity and pagerank was proposed (Sathiyakumari & Vijaya, 2018). another overlapping community detection algorithm for directed networks which uses a Gamma-Poisson block model was introduced. The model can also be generalized for undirected networks by means of making the block model matrix as symmetric (Gao, Liu, & Miao, 2018). A multi-objective optimization model for clustering the heterogeneous weighted networks through key nodes identification with overlapping communities was introduced (Kalantari, Ghazanfari, Fathian, & Shahanaghi, 2020).

Some approaches are based on nature inspired algorithms. a consensus genetic based algorithm was used to detect communities in directed networks (Mathias, Rosset, & Nascimento, 2016). In another work Bio-inspired algorithms was used for detecting communities in weighted directed networks (Osaba et al., 2018). An ant colony based algorithm for overlapping community detection was introduced (X. Zhou, Liu, Zhang, Liu, & Zhang, 2015). Although Optimization of an objective function and using nature inspired algorithms can be classified in one category, we'd rather to split them in two different categories because of uniqueness of problem solving approach in nature inspired algorithms. So far the most of the algorithms under study did not take the effect of edge weights into account. A new local clustering coefficient is proposed for weighted and directed networks which captures the presence of triangles as well as weights (Clemente & Grassi, 2018). An algorithm in which impact factors of in-degree and out-degree are considered and the directed weighted degree is used to measure the importance of a node (Liu, Qin, Yun, & Wu, 2011). In Table 1 we summarized the algorithms based on the method they have used to tackle the community detection problem in weighted and directed networks. As it can be shown, the number of algorithms that took edge weight into consideration in the algorithm is too low. It is important

to mention that our algorithm is an extension to the algorithm proposed in (Xing, Fanrong, Yong, & Ranran, 2015). This algorithm is designed for undirected and unweighted networks. We extended the algorithm for weighted and directed networks, meanwhile we developed a new modularity function called Weighted Flow Measure (WFM) fitted for weighted and directed graphs and a new overlapping score which considers the similarity between edges and nodes of two given communities at the same time.

Table 1
A brief review on community detection algorithms in weighted and directed networks based on the method, community type and network type

Method and process					Communities	Network	Paper
Nature inspired	Objective function	Converting to bipartite	Directionality to weights	naive			
□	□	□	■	□	Overlapping	Directed Unweighted	(Long & Li, 2017)
□	■	□	□	□	Disjoint	Directed Unweighted	(Tosyali et al., 2019)
■	□	□	□	□	Disjoint	Directed Unweighted	(Mathias et al., 2016)
■	□	□	□	□	Disjoint	Weighted Directed	(X. Zhou et al., 2015)
■	□	□	□	□	Overlapping	Directed Unweighted	(Osaba et al., 2018)
□	■	□	□	□	Overlapping	Directed Unweighted	(Santos et al., 2016)
□	■	□	□	□	Overlapping	Directed Unweighted	(Sathiyakumari & Vijaya, 2018)
□	■	□	□	■	Overlapping	Directed Undirected Unweighted	(Gao et al., 2018)
□	□	■	□	□	Disjoint	Directed Unweighted	(Guimerà et al., 2007)
□	□	■	□	□	Disjoint	Directed Unweighted	(D. Zhou, Huang, & Schölkopf, 2005)
□	■	□	□	□	Disjoint	Directed Weighted	(Clemente & Grassi, 2018)
□	■	□	□	□	Disjoint	Directed Weighted	(Liu et al., 2011)
□	■	□	□	□	Overlapping	Undirected Weighted	(Kalantari et al., 2020)
□	■	□	□	□	Overlapping	Directed Weighted	This Work

3. Model

We use the graph $G=(V,E)$ to model the weighted and directed network in which V is the set of nodes and E is the set of weighted directed edges between the nodes. As it can be seen from Fig. 1, In this model we normalize the edge weights by dividing each edge weight to the maximum edge weight.

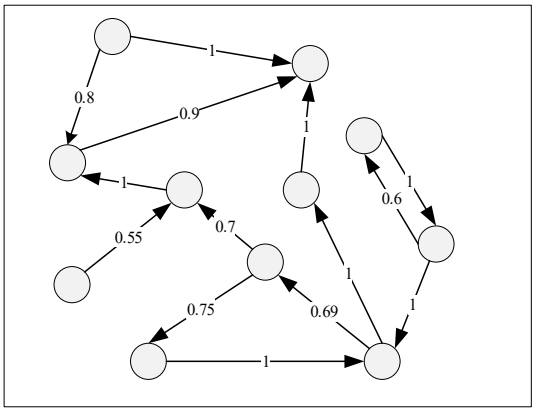

Fig. 1. The network model under study

4. Algorithm

In this section we present the algorithm. The algorithm is designed for overlapping community detection in weighted and directed networks and is called LCFE. First we present the notations used in the algorithm pseudocode.

4.1 LCFE pseudocode notations

The notations and the parameter definition of the LCFE can be seen in Table 2.

Table 2
Parameter notations used in the algorithm

Parameter	Description
$G=\langle V,E\rangle$	A network in which V denotes the set of vertices and E represents the weighted and directed edges
$A_{n\times m}$	Adjacency matrix of the graph which can be asymmetric due to the existence of directed edges
LC	The set of local communities
$Com(i)$	Set of communities which node i belong to
$N(i)$	Neighbors of the node i
$N(C_i)$	The communities which have common nodes with community C_i
V_i	Set of nodes in the community i
E_i^{in}	Set of inner edges in community i
$NC(i)$	Communities which contain the neighbors of node i
$\mathcal{C}$	Set of final communities
$\theta(v_i, v_j)$	A binary parameter. 1 when v_i and v_j both belong to a community and 0 otherwise.
$\lambda(v_i, v_j)$	A binary parameter. 1 if only on of v_i and v_j belong to a community.
ψ_c	Summation of edge weights of a closed cycle in a community.
$\rho_{u,v}(e)$	Number of shortest paths that connect nodes u and v by crossing edge e
$\rho_{u,v}(e)$	Number of shortest paths that connect nodes u and v
$B(e)$	Centrality of edge e

4.2 LCFE Steps

LCFE begins with certain edges, sets the ends of the edge as local communities and expands the local communities. In the second step the local communities are expanded through a modularity function optimization. In the third step the local communities are merged based on a certain criterion. And finally in the fourth step the communities are refined through assigning the homeless nodes to detected communities.

4.2.1 Step1: calculating the edge betweenness centrality

First step is calculating the edge betweenness of all edges in the graph by considering the weights. The more central an edge the stronger the communities developed from it. This step is a new contribution to (Xing et al., 2015) because the edges are not chose randomly in order to community expansion. The centrality criteria is the one developed in (Girvan & Newman, 2002):

$$B(e)=\sum_{\{u,v\}\in\binom{V}{2}}\frac{\rho_{u,v}(e)}{\sigma_{u,v}} \tag{1}$$

As we know, the weight affects the numbers of shortest paths between two nodes, so in order to relax the negative effect on the intensity concept i.e. edge weights we sort the edge betweenness values in ascending order. Here is the pseudocode of this step:

Pseudocode 1: Edge Betweenness matrix

Step 1: Calculating the edge betweenness Centrality Matrix

Input: Network $G=\langle V,E\rangle$

Output: Edges Sorted by value of betweenness $BM_{|V|\times|V|}$

for $v_i \in \{V\}$
  for $v_{j\neq i} \in \{V\}$
    $B(e)=\sum_{\{v_i,v_j\}\in\binom{V}{2}}\frac{\rho_{v_i,v_j}(e)}{\sigma_{v_i,v_j}}$
  end for
end for
sort BM in ascending order

4.2.2 Step 2: Local communities

After calculating edge betweenness for each edge, the algorithm starts by the edges with the first edge in BM and assigns its ends as the local community, then the common neighbors of the ends will be drawn out. After that a new modularity function will determine whether a common neighbor node will be joined to the community or not.

Definition 1. Weighted Flow Modularity (WFM):

This modularity function is mostly based on the popular M function modularity. But due to the existence of directionality and the assumption of stronger communities in directed graphs have stronger information cycles, we extended the M function as following:

$$WFM = \frac{M_{\text{int}ernal}}{M_{external}} + \psi_c = \frac{\sum_{v_i v_j} A_{ij}.\theta\left(v_i, v_j\right)}{\sum_{v_i v_j} A_{ij}.\lambda\left(v_i, v_j\right)} + \sum_{ij \in \psi_c} A_{ij} \tag{2}$$

In the Eq. (3), $M_{\text{int}ernal}$ is the summation of edge weights that are inside a community and $M_{external}$ is the summation of edge weights that are outside of a community. Finally ψ_c is Summation of edge weights of a closed cycle in a community. A common neighbor which makes a closed cycle starting from itself and ending to itself will have more chance to join the local community because it increases the flow of information to the existing nodes of local community and itself. After clarification on the WFM, it is time to scrutinize the pseudocode for the local community detection step of the algorithm. Here is the pseudocode:

Pseudocode 2: Local Community Detection

Step 2: Detecting Local Communities

Input: Network $G = \langle V, E \rangle$, $BM_{|V|\times|V|}$, $LC = \{\}$

Output: local communities $LC = \{LC_1, LC_2, ..., LC_k\}$

```
for e ∈ BM & e ∉ IE
  LC_temp = {a, b}_e
  if Com(a) ∩ Com(b) = φ & {a, b}_e ∉ LC
    NC = N(a) ∩ N(b) − {a, b}
    for {v} ∈ NC
      if WFM(LC_temp ∪ v) ≥ WFM(LC_temp)
        LC_temp = LC_temp ∪ {v}
        IE = IE ∪ e(LC_temp)
      end if
    end for
  end if
  LC = LC ∪ LC_temp
end for
```

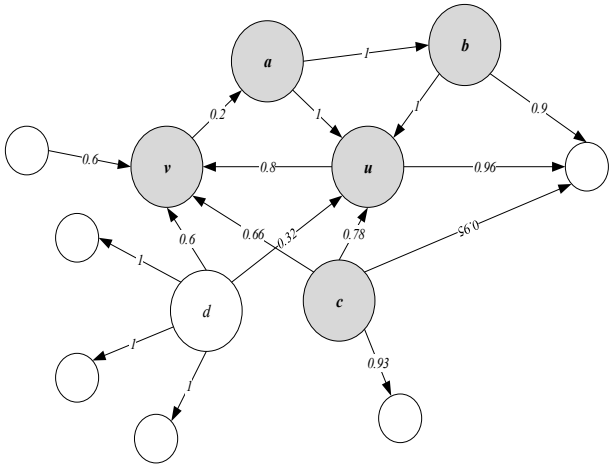

Fig. 2. Local Community Expansion

For better clarification of this step, consider the following example using Fig. 2. Suppose e_{uv} is the starting edge probed from step 1. Nodes u and v make up the first local community. WFM for this community is equal to 0.13. On the other hand the neighbor nodes set is $\{a, b, c, d\}$. Now every node in the neighboring set will be added to the local community. If it increases the amount of WFM, then it will be added to the community. First candidate is node a, WFM value for local community with a is equal to 2.33 which is larger than the amount of WFM before joining a. so the local community will be $\{u, v, a\}$. It can be seen that the significant increase in the amount of WFM is due to existence of a closed cycle starting from a. If the b is added to the local community the value of WFM with b will be 5.82 which is larger than the value of WFM for the community before joining b, so node b will be added to the local community. By adding node d, the value of WFM will be 5.78 which is smaller than the previous amount. It can be seen that node d does not initiate a closed cycle of information with the current nodes of the local community. Finally node c will be added to the local community. We can see that the amount of WFM by joining node c will be rised to 6.24. So node c will be added to the local community. At the end of this process we can see that the local community expanded to $\{u, v, a, c, b\}$. This step of the algorithm continues till all of the edges has been investigated.

4.2.3. Step 3: Local Community Merging

Extracted local communities from step 2 are relatively small and cannot be considered as the final community structure of the network. On the other hand they are not so much overlapping. So in this step we developed a novel overlapping score in which considers the overlapping in nodes and edges at the same time. Before that we will have a review on the base overlapping scores on which we developed our own:

Definition 2. Overlapping score (OS) was first introduced in (Nguyen, Dinh, Nguyen, & Thai, 2011). This score is parameter free and requires only the local topological information of the network. Here is the equation:

$$OS(C_i, C_j) = \frac{\left|V_i \cap V_j\right|}{\min\left\{\left|V_i\right|, \left|V_j\right|\right\}} + \frac{\left|E_i^{in} \cap E_j^{in}\right|}{\min\left\{\left|E_i^{in}\right|, \left|E_j^{in}\right|\right\}} \tag{3}$$

Definition 3. Later on, the equation (3) was extended in (Xing et al., 2015) to the following form:

$$WOS(C_i, C_j) = \alpha \frac{\left|V_i \cap V_j\right|}{\min\left\{\left|V_i\right|, \left|V_j\right|\right\}} + (1-\alpha) \frac{\left|E_i^{in} \cap E_j^{in}\right|}{\min\left\{\left|E_i^{in}\right|, \left|E_j^{in}\right|\right\}} \tag{4}$$

In this new form, a parameter α is added to the fraction because some networks have more overlapping score in nodes rather than edges.

Definition 4. A similarity index was introduced in (Carley, 1991). The main idea behind this similarity index is that "Friends tend to be similar". The equation describing this index is as following:

$$similarity(A, B) = \sum_{SharedItems} \frac{1}{\log[frequency(sharedItem)]} \tag{5}$$

Definition 5. Based on what have reviewed, we developed a novel overlapping score called logarithmic overlapping score (LOS). The equation for LOS is as follows:

$$LOS(C_i, C_j) = \alpha \frac{1}{\log(|V_i \cap V_j|)} + (1-\alpha) \frac{1}{\log(|E_i^{in} \cap E_j^{in}|)} \tag{6}$$

LOS is tunable through different values of α.

After illustration on LOS, the pseudocode of step 3 is as following:

Pseudocode 3: Community Merging

Step 3: local community merging

Input: Local communities $LC = \{LC_1, LC_2, ..., LC_k\}$

Output: Communities $C = \{C_1, C_2, ..., C_n\}$

```
C = LC
Tabu_list = {}
for C_i ∈ C
    if C_i ∉ Tabu_list
        C = C / C_i
            for C_j ∈ C
                if C_j ∉ Tabu_list and LOS(C_i, C_j) ≥ β
                    Union(C_i) = {C_i, C_j}
                    C_i = Union(C_i) ∪ C_i
                    C' = C' / Union(C_i)
                    Tabu_list = Tabu_list ∪ Union(C_i)
                end if
            end for
    end if
    for u ∈ C_i
        update Com(u)
    end for
    C' = C' / C_i
end for
```

In the pseudocode above, β is a tunable parameter. The larger the value of β the less communities combined.

Step 4: Community Refinement

After merging communities there might be nodes that are left without communities. Now that the communities are large enough, the possibility of these nodes to join a community is higher because they have a better chance to form closed cycles. At the end of this step every node that hasn't joined a community is called an outlier.
In order for a node which is out of community to join a community we define a criteria called node fitness:

Definition 6. The value of fitness for a node as calculated as following:

$$Fitness(C,\{node\}) = WFM_{C\cup\{node\}} - WFM_{C-\{node\}} \tag{7}$$

By joining the node to the community C, if the fitness value us strictly larger than 0, then node will be joined to the community. The pseudocode of this step is as following:

Pseudocode 4: Community Refining

Step 4: Community refining
Input: Merged Communities from step 3 **Output:** Final Community structure of the network 02. $Outlier = \{\}$ 03. **for** $u \in V$ *and* $Com(u) = \phi$ 04. **for** $C_i \in NC(u)$ 05. **if** $Fitness(C_i,\{u\}) > 0$ 06. $C_i = C_i \cup \{u\}$ 07. $Com(u) = Com(u) \cup \{i\}$ 08. **end if** 09. **end for** 10. **if** $Com(u) = \phi$ 11. $Outlier = Outlier \cup \{u\}$ 12. **end if** 13. **end for**

5. Numerical example and benchmarking

5.1 Algorithm for Generating Benchmark networks

For the purpose of benchmarking we used the algorithm introduced in (Lancichinetti & Fortunato, 2009). The algorithm is specifically designed for testing overlapping community detection algorithms in weighted and directed networks.

5.2 Run time settings and environment

The simulations have been carried out on a laptop with Intel(R) Core(TM) i5 m48 @ 2.67GHz 2.66 GHz processor and 3.87 GB Memory under Win8 operating system. The source code of the algorithm of this article is written in Python 3.7. The benchmark algorithm has its own software package developed.

5.3 Evaluation Criteria of the LCFE

We will test the algorithm performance with normalized mutual information (NMI) for benchmark networks since the community structure of the LFR networks are already known and EQ measure for the real world networks. Considering the fact that, the true community structure of most real networks is unknown, we utilize the EQ measure to evaluate the performance of the algorithms. This measure is calculated through Eq. (8):

$$EQ = \frac{1}{2m}\sum_{c=1}^{k}\sum_{u,v\in C_c}\frac{1}{O_u O_v}\left(A_{uv} - \frac{d_u d_v}{2m}\right) \tag{8}$$

m is the number of edges; O_u and O_v are the number of communities that incorporate the nodes u and v; $A_{u,v}$ shows the adjacency. The greater the EQ value is, the better community detection result.

5.4 Parameters used in benchmark networks

The mentioned algorithm for generating benchmark networks gets the parameters shown in Table 3 as inputs and generates a network based on the input values.

Table 3
Benchmark generator algorithm parameters

Parameter	Description		
N	Number of nodes	t1	Minus exponent for the degree sequence
k	Average degree	t2	Minus exponent for the community distribution
maxk	Maximum degree	minc	Minimum community size
mut	Mixing parameter for topology	maxc	Maximum community size
Muw	Mixing parameter for weights	on	Number of overlapping nodes
Beta	Exponent for weight distribution	om	Number of memberships for overlapping nodes

5.5 Experimental results on synthetic networks

In this section, we evaluate the algorithm from two points of view. The first one for the accuracy and the second one for algorithm run time.

5.5.1 Experiments for accuracy evaluation

In this section, we generated 64 synthetic networks divided into 16 groups each containing 4 networks. All groups share the common parameters N, k, maxk, beta, t1 and t2. All networks in each group share all parameters except for the number of overlapping nodes. We extracted the community structures with LCFE algorithm and Order Statistics Local Optimization Method OSLOM (Lancichinetti, Radicchi, Ramasco, & Fortunato, 2011). In Table 6 the synthetic network groups for accuracy evaluation of LCFE is listed. The comparison results are shown in Fig. 2. It can be seen from Fig. 3 to Fig. 18t hat the LCFE is dominant over OSLOM in most cases.

5.5.2 LCFE run time evaluation

In this part of the evaluation process, we generated 10 different benchmark networks. From 100 nodes to 1000 nodes. It can be seen from Fig. 19 that LCFE is quite better than OSLOM in most cases and is competitive in other cases.

5.6 Experimental results on Real world networks

We implemented the LCFE on 10 real world weighted and directed networks from KONECT project. A brief description of these networks can be seen in Table 4. We used EQ measure in order to evaluate the quality of the community detection that carried out by LCFE and OSLOM. The more the EQ measure, the better the community detection result. EQ measure evaluation results on real world networks are listed in Table 5.

Table 4
Real World Networks

Network	Edges	Nodes
Adolescent health	12969	2539
High School	366	70
Residence Hall	2672	217
Seventh Grades	376	29
US Airports	28236	1504
Florida Ecosystem Dry	2137	128
Florida Ecosystem Wet	2106	127
Macaques	1187	62
Bison	314	26
Rhesus	111	16

Table 5
EQ measure Comparison between OSLOM and LCFE on 10 Real World Networks

Network	OSLOM	LCFE
Adolescent health	0.369	0.4125
High School	0.7452	0.7698
Residence Hall	0.6341	0.5269
Seventh Grades	0.325	0.374
US Airports	0.485	0.436
Florida Ecosystem Dry	0.6635	0.721
Florida Ecosystem Wet	0.6896	0.7136
Macaques	0.8526	0.6923
Bison	0.3654	0.5498
Rhesus	0.3699	0.4779

6. Discussion and conclusion

In this article, we developed an overlapping community detection algorithm for weighted and directed networks. The main contributions of this work are using sorted edges as initiators of community detection process, developing a new modularity function called weighted flow modularity based on M function and weighted flow cycles, a new overlapping score which considers overlapping between nodes and edges at the same time. We generated 81 LFR benchmark in order to evaluate the various aspects of the developed algorithm in terms of accuracy, run time and parameter selection. We evaluated the community detection results on LFR benchmarks using normalized mutual information. Then the performance of the algorithm was evaluated using EQ measure on real world networks. In all cases, the performance of the algorithm was compared with Order Statistical Optimization Method (OSLOM). LCFE was dominant over OSLOM in most cases and was competitive in other cases.

Table 6
16 Groups of LFR Networks

	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12	N13	N14	N15	N16
N	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
K	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
MAXK	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25
MUT	0.1	0.1	0.1	0.1	0.3	0.3	0.3	0.3	0.1	0.1	0.1	0.1	0.3	0.3	0.3	0.3
MUW	0.2	0.2	0.4	0.4	0.2	0.2	0.4	0.4	0.2	0.2	0.4	0.4	0.2	0.2	0.4	0.4
BETA	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
T1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
T2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
MINC	5	5	5	5	5	5	5	5	10	10	10	10	10	10	10	10
MAXC	25	25	25	25	25	25	25	25	50	50	50	50	50	50	50	50
OM	2	5	2	5	2	5	2	5	2	5	2	5	2	5	2	5
ON	0-75	0-75	0-75	0-75	0-75	0-75	0-75	0-75	0-75	0-75	0-75	0-75	0-75	0-75	0-75	0-75

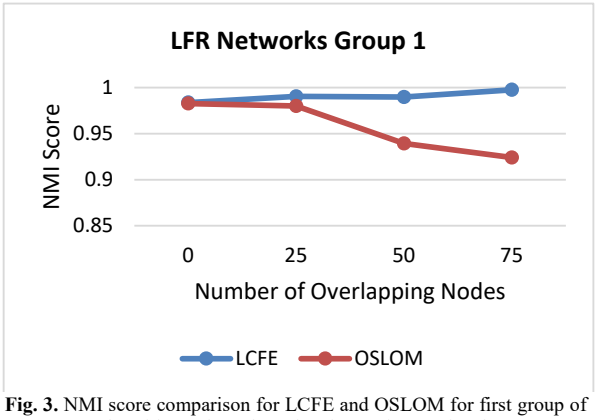

Fig. 3. NMI score comparison for LCFE and OSLOM for first group of LFR Networks with number of overlapping nodes ranging from 0 to 75

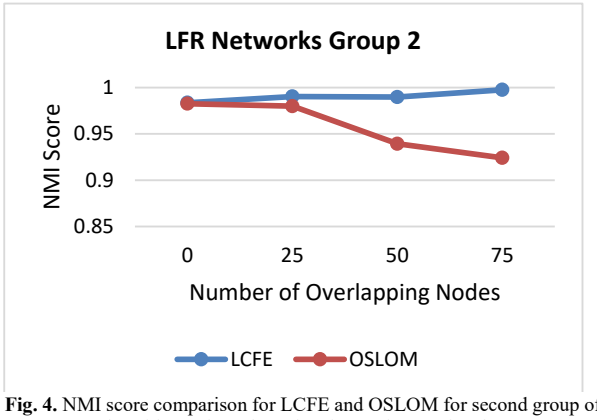

Fig. 4. NMI score comparison for LCFE and OSLOM for second group of LFR Networks with number of overlapping nodes ranging from 0 to 75

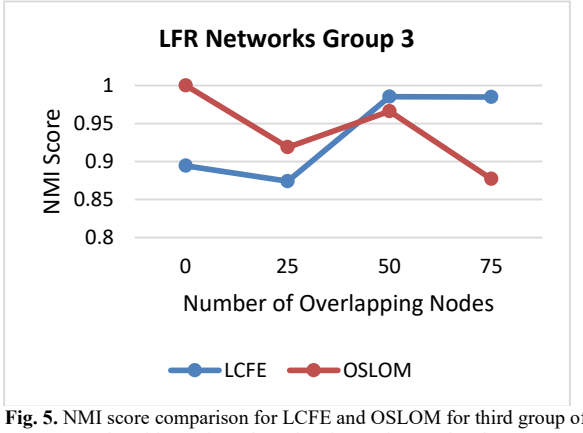

Fig. 5. NMI score comparison for LCFE and OSLOM for third group of LFR Networks with number of overlapping nodes ranging from 0 to 75

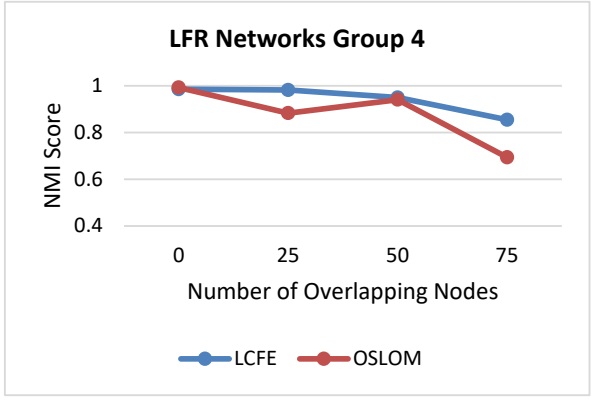

Fig. 6. NMI score comparison for LCFE and OSLOM for fourth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

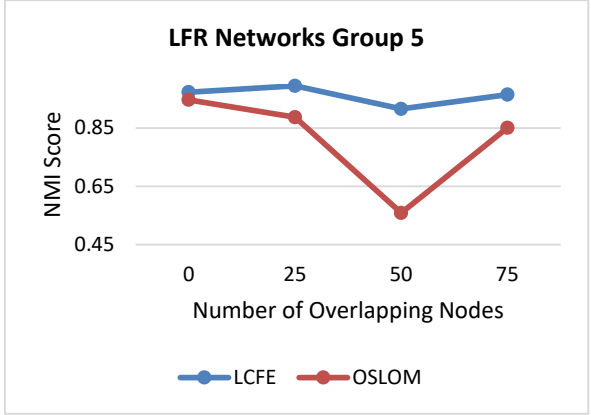

Fig. 7. NMI score comparison for LCFE and OSLOM for fifth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

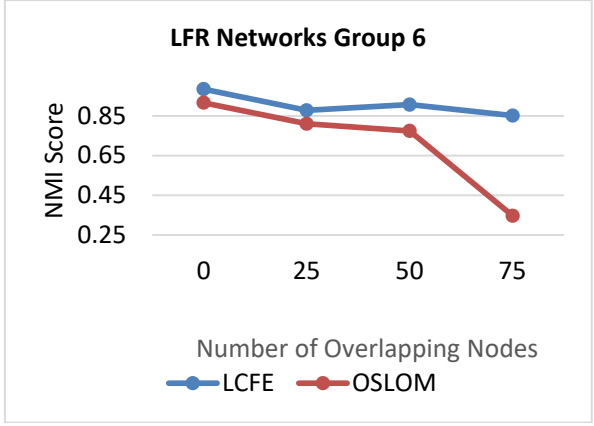

Fig. 8. NMI score comparison for LCFE and OSLOM for sixth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

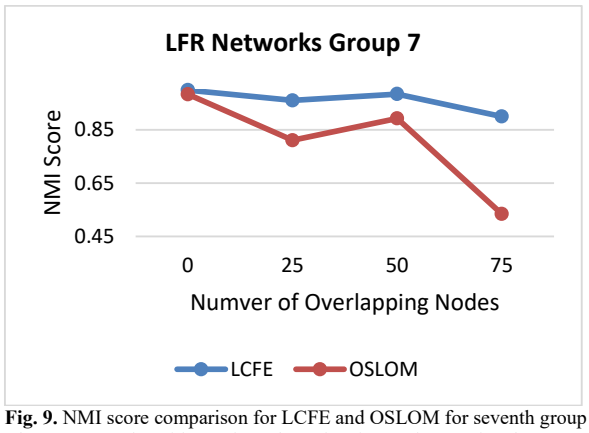

Fig. 9. NMI score comparison for LCFE and OSLOM for seventh group of LFR Networks with number of overlapping nodes ranging from 0 to 75

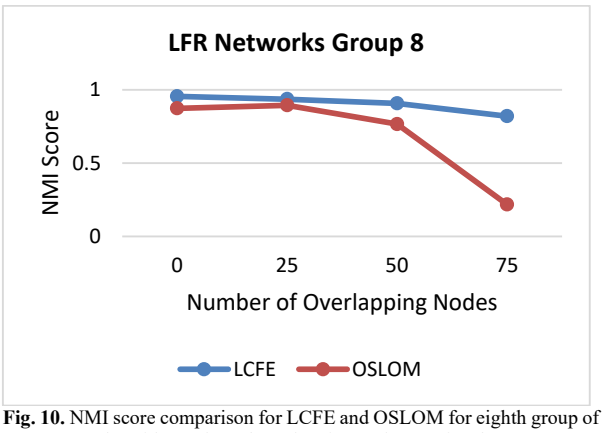

Fig. 10. NMI score comparison for LCFE and OSLOM for eighth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

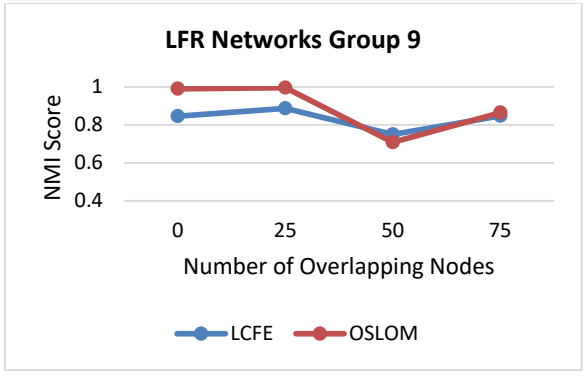

Fig. 11. NMI score comparison for LCFE and OSLOM for nineth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

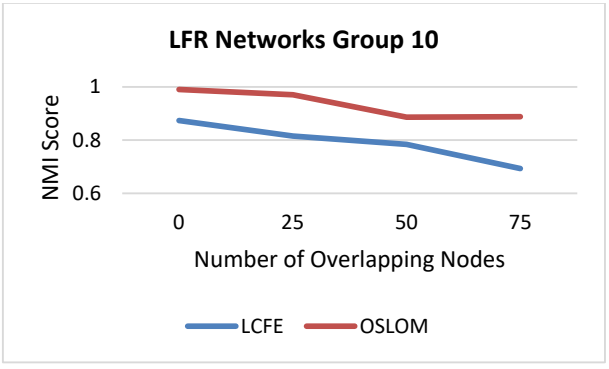

Fig. 12. NMI score comparison for LCFE and OSLOM for tenth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

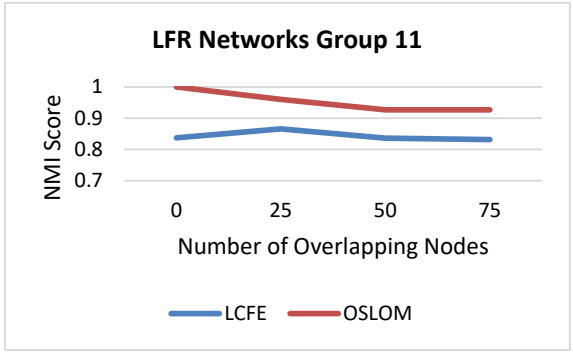

Fig. 13. NMI score comparison for LCFE and OSLOM for eleventh group of LFR Networks with number of overlapping nodes ranging from 0 to 75

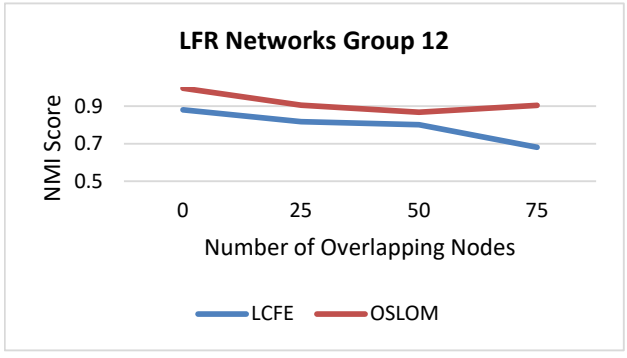

Fig. 14. NMI score comparison for LCFE and OSLOM for twelveth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

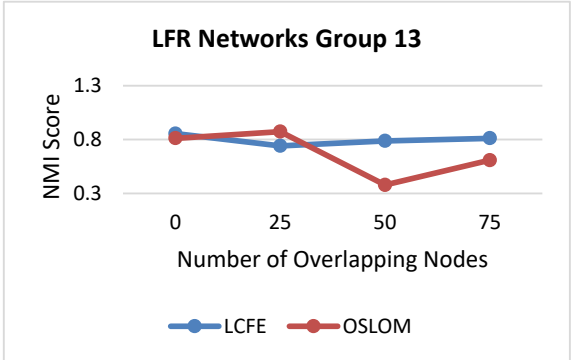

Fig. 15. NMI score comparison for LCFE and OSLOM for thirteenth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

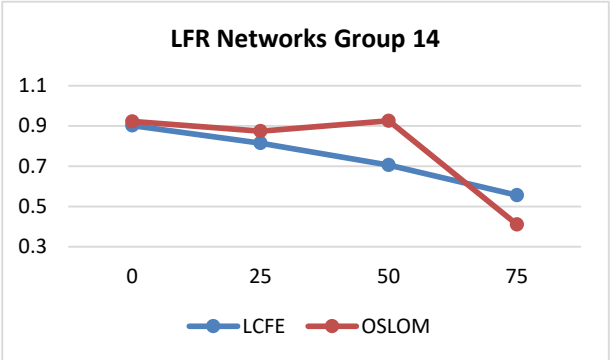

Fig. 16. NMI score comparison for LCFE and OSLOM for fourteenth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

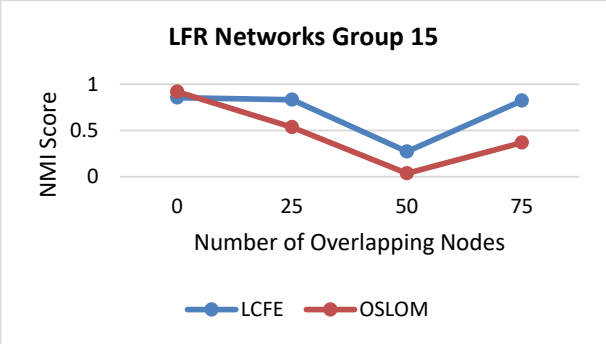

Fig. 17. NMI score comparison for LCFE and OSLOM for fifteenth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

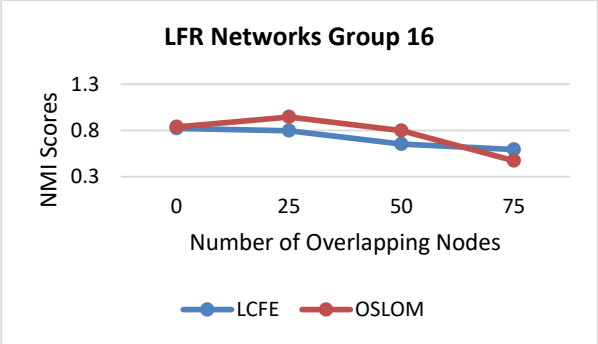

Fig. 18. NMI score comparison for LCFE and OSLOM for sixeenth group of LFR Networks with number of overlapping nodes ranging from 0 to 75

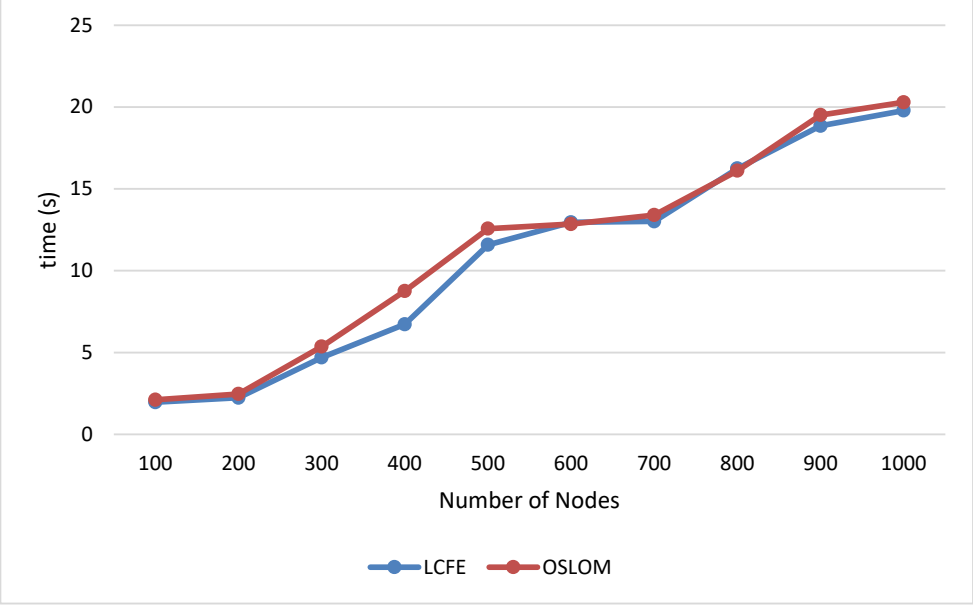

Fig. 19. Run Time Analysis of LCFE and OSLOM

References

Arenas, A., Duch, J., Fernández, A., & Gómez, S. (2007). Size reduction of complex networks preserving modularity. *New Journal of Physics, 9*.

Badiee, A., & Ghazanfari, M. (2018). A monopoly pricing model for diffusion maximization based on heterogeneous nodes and negative network externalities (Case study: A novel product). *Decision Science Letters, 7*(3), 287-300.

Carley, K. (1991). A theory of group stability. *American Sociological Review, 56*(3), 331-354. doi:10.2307/2096108

Clemente, G. P., & Grassi, R. (2018). Directed clustering in weighted networks: A new perspective. *Chaos, Solitons & Fractals, 107*, 26-38.

Dugué, N., & Perez, A. (2015). *Directed Louvain : maximizing modularity in directed networks*.

Gao, S., Liu, R., & Miao, H. (2018, 28-30 July 2018). *A Gamma-Poisson Block Model for Community Detection in Directed Network.* Paper presented at the 2018 14th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD).

Girvan, M., & Newman, M. E. (2002). Community structure in social and biological networks. *Proceedings of the national academy of sciences, 99*(12), 7821-7826.

Guimerà, R., Sales-Pardo, M., & Amaral, L. A. N. (2007). Module identification in bipartite and directed networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics, 76*(3). doi:10.1103/PhysRevE.76.036102

Kalantari, H., Ghazanfari, M., Fathian, M., & Shahanaghi, K. (2020). Multi-objective optimization model in a heterogeneous weighted network through key nodes identification in overlapping communities. *Computers & Industrial Engineering, 144*, 106413.

Kermani, M. A. M. A., Badiee, A., Aliahmadi, A., Ghazanfari, M., & Kalantari, H. (2016). Introducing a procedure for developing a novel centrality measure (Sociability Centrality) for social networks using TOPSIS method and genetic algorithm. *Computers in Human Behavior, 56*, 295-305.

Kim, Y., Son, S. W., & Jeong, H. (2010). Finding communities in directed networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics, 81*(1). doi:10.1103/PhysRevE.81.016103

Lai, D., Lu, H., & Nardini, C. (2010a). Extracting weights from edge directions to find communities in directed networks. *Journal of Statistical Mechanics: Theory and Experiment, 2010*(6). doi:10.1088/1742-5468/2010/06/P06003

Lai, D., Lu, H., & Nardini, C. (2010b). Finding communities in directed networks by PageRank random walk induced network embedding. *Physica A: Statistical Mechanics and its Applications, 389*(12), 2443-2454. doi:10.1016/j.physa.2010.02.014

Lancichinetti, A., & Fortunato, S. (2009). Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical review. E, Statistical, nonlinear, and soft matter physics, 80*, 016118. doi:10.1103/PhysRevE.80.016118

Lancichinetti, A., Radicchi, F., Ramasco, J. J., & Fortunato, S. (2011). Finding statistically significant communities in networks. *PloS one, 6*(4), e18961-e18961. doi:10.1371/journal.pone.0018961

Leicht, E. A., & Newman, M. E. J. (2008). Community structure in directed networks. *Physical Review Letters, 100*(11). doi:10.1103/PhysRevLett.100.118703

Liu, H., Qin, X., Yun, H., & Wu, Y. (2011, 2011//). *A Community Detecting Algorithm in Directed Weighted Networks.* Paper presented at the Electrical Engineering and Control, Berlin, Heidelberg.

Long, H., & Li, B. (2017). Overlapping Community Identification Algorithm in Directed Network. *Procedia Computer Science, 107*, 527-532. doi:https://doi.org/10.1016/j.procs.2017.03.105

Malliaros, F. D., & Vazirgiannis, M. (2013). Clustering and community detection in directed networks: A survey. *Physics reports, 533*(4), 95-142. doi:10.1016/j.physrep.2013.08.002

Mathias, S. B. B. R. P., Rosset, V., & Nascimento, M. C. V. (2016). Community Detection by Consensus Genetic-based Algorithm for Directed Networks. *Procedia Computer Science, 96*, 90-99.

Newman, M. E. J., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics, 69*(2 2), 026113-026111-026113-026115. doi:10.1103/PhysRevE.69.026113

Nguyen, N. P., Dinh, T. N., Nguyen, D. T., & Thai, M. T. (2011, 9-11 Oct. 2011). *Overlapping Community Structures and Their Detection on Social Networks.* Paper presented at the 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing.

Osaba, E., Del Ser, J., Camacho, D., Galvez, A., Iglesias, A., Fister, I., & Fister, I. (2018, 2018//). *Community Detection in Weighted Directed Networks Using Nature-Inspired Heuristics.* Paper presented at the Intelligent Data Engineering and Automated Learning – IDEAL 2018, Cham.

Santos, C. P., Carvalho, D. M., & Nascimento, M. C. V. (2016). A consensus graph clustering algorithm for directed networks. *Expert Systems with Applications, 54*, 121-135.

Sathiyakumari, K., & Vijaya, M. S. (2018, 2018//). *Identification of Subgroups in a Directed Social Network Using Edge Betweenness and Random Walks.* Paper presented at the Smart Computing and Informatics, Singapore.

Satuluri, V., & Parthasarathy, S. (2011). *Symmetrizations for clustering directed graphs.* Paper presented at the ACM International Conference Proceeding Series.

Tosyali, A., Kim, J., Choi, J., & Jeong, M. K. (2019). Regularized asymmetric nonnegative matrix factorization for clustering in directed networks. *Pattern Recognition Letters, 125*, 750-757.

Xing, Y., Fanrong, M., Yong, Z., & Ranran, Z. (2015). Overlapping Community Detection by Local Community Expansion. *Journal of Information Science and Engineering, 31*, 1213-1232.

Zhou, D., Huang, J., & Schölkopf, B. (2005). Learning from labeled and unlabeled data on a directed graph. *Proceedings of the 22nd International Conference on Machine Learning*, 1036-1043.

Zhou, D., Schölkopf, B., & Hofmann, T. (2005). Semi-supervised learning on directed graphs. *Advances in Neural Information Processing Systems*, 1633-1640.

Zhou, X., Liu, Y., Zhang, J., Liu, T., & Zhang, D. (2015). An ant colony based algorithm for overlapping community detection in complex networks. *Physica A: Statistical Mechanics and its Applications, 427*, 289-301.