A Fast Algorithm for Overlapping Community Detection

Mostafa Elyasi, Mohammadreza Meybodi
Department of Computer Engineering & IT
Amirkabir University of Technology (Tehran Polytechnic)
Tehran, Iran
{m.elyasi, mmeybodi}@aut.ac.ir

Alireza Rezvanian, Maryam Amir Haeri
Department of Computer Engineering & IT
Amirkabir University of Technology (Tehran Polytechnic)
Tehran, Iran
{ a.rezvanian, haeri}@aut.ac.ir

Abstract— **Nowadays, the emergence of online social networks have empowered people to easily share information and media with friends. Interacting users of social networks with similar users and their friends form community structures of networks. Uncovering communities of the online users in social networks plays an important role in network analysis with many applications such as finding a set of expert users, finding a set of users with common activities, finding a set of similar people for marketing goals, to mention a few. Although, several algorithms for disjoint community detection have been presented in the literature, online users simultaneously interact with their friends having different interests. Also users are able to join more than one group at the same time which leads to the formation of overlapping communities. Thus, finding overlapping communities can realize a realistic analysis of networks. In this paper, we propose a fast algorithm for overlapping community detection. In the proposed algorithm, in the first phase, the Louvain method is applied to the given network and in the second phase a belonging matrix is updated where an each element of belonging matrix determines how much a node belongs to a community. Finally, some of the found communities are merged based on the modularity measure. The performance of the proposed algorithm is studied through the simulation on the popular networks which indicates that the proposed algorithm outperforms several well-known overlapping community detection algorithms.**

Keywords-community detection; overlapping communities; social network analysis; belonging matrix; larg networks; fast algorithm.

I. INTRODUCTION

In the recent years, development of internet and emergence of online social networks (OSNs) have been affected on the daily lives of many people. OSNs have attracted people all around the world and now OSNs are the biggest source of various type of information. Lots of scientists and researchers are interested in analyzing these sources. Researchers are trying to find patterns and discovering the structures of networks. The goal is finding the people who have same behaviors by extracting their features and putting them in same groups. Researchers have been devised several methods to find group or cluster of members in network. Usually, social network represented with a set of nodes which represents users and a set of edges which represents a typical connection between users.

Community detection refers to find a set of nodes (also known as group, module, cluster or community) where nodes of within a community have a more similarity than that of other nodes outside the community. Thus, nodes in a community are necessarily denser than that of other nodes outside the community. The community detection algorithms have been presented in the literature in order to discover the community structures of the network. However, there have been several challenges for community detection algorithms. For instance, the number and the size of the communities are unknown and also in real world, size of networks are massive and the structure of network are dynamically is changed with time. Due to the significant applications of community detection, several community detection approaches have been presented in the literature which can be classified into six categories: spectral and clustering methods, hierarchical algorithms, modularity-based methods, model-based methods, local community detection methods, and feature-based assisted methods [1]. There are some stochastic models like Bayesian principle model [2] and stochastic block model (SBM) [3] for detecting the communities. SBMs are model based algorithms and probabilistic tools explaining the connectivity relationship between pairs of nodes. Another famous algorithms for improving *Bayesian* methods are Order Statistics Local Optimization Method (OSLOM) [4] by merging communities and *Pareto* optimality method [5]. The accuracy of community detection methods is usually evaluated by adjusted rand indices [6] and Normalized Mutual Information (NMI) [7]. These measures compute the similarity between two data clustering methods. In is noted that these evaluation methods can be used when there exists a grand truth for evaluation. On the contrary, for real networks, there is no any grand truth and for this reason some researchers presented different objective function for evaluation such as modularity measure which is presented by *Newman* [8]. Due to large size of real networks, scalable algorithms are very beneficial for social network analysis. One of the most popular efficient and very fast community detection algorithm is the *Louvain* method [9] which works based on modularity increasing.

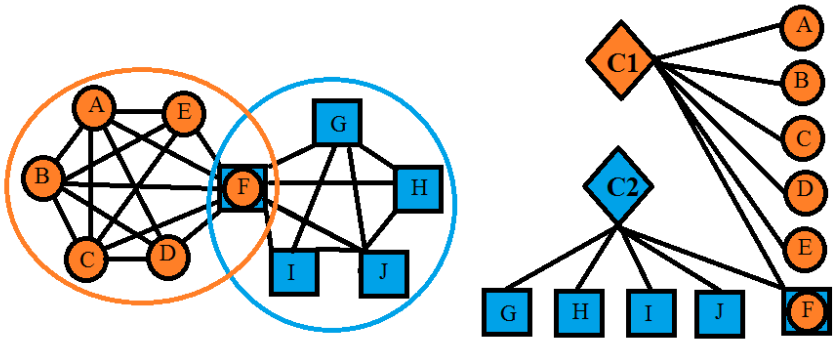


Figure 1. Overlapping in communities

Sometimes, users of social network may not be part of only a community but also it may be a member of many similar or different communities among the network. For example, a researcher concurrently may collaborate with some researchers from several universities. Therefore, the network of collaborations between researchers has a member in two communities as is the case with node F in Fig. 1. Hence, recently new methods have been designed to discover overlapping communities which are more realistic. Such an example for overlapping communities is clique percolation method (CPM) [10] as presented the output of this algorithm in Fig. 1. In this algorithm, fully connected nodes in each community have been imagined by CPM, which these fully connected subgraphs named cliques. Moreover, if two cliques have shared some nodes, these nodes will belong to both communities, that mean the communities are overlapped. Another approach to overlapping community detection is based on modularity maximization. *Gregory et al.* proposed an algorithm that is an extension of Girvan-Newman [11] clustering method, he called it CONGA. CONGA uses Girvan-Newman method but it allows a node to split into multiple nodes and each copy can stay in different community than the original is. CONGA uses node centrality and edge betweenness to detect communities. Also, Gregory improved his method by using local betweenness and named it CONGO [12] which is faster than CONGA. COPRA [13] is another method has been proposed by Gregory. COPRA using label propagation based algorithm, allows each node to have multiple labels. Similarly, *Xie* et al. [14] proposed speaker-listener label propagation algorithm (SLPA). In SLPA each node can hold multi labels during the running of the algorithm. Besides, evolutionary algorithms such as genetic algorithms are used for community detection. These methods as an iterative algorithm with fitness function try to optimize the fitness value. As an example, GA-Net [15] uses a new efficient function and optimizes the fitness value to decreasing the problem space and find dense communities.

In this paper, we propose a fast algorithm for overlapping community detection. In the proposed algorithm, at the first phase the Louvain method is applied to the network and in the second phase a belonging matrix is updated where an each element of the belonging matrix determines how much a node belongs to a community. Finally, some of the found communities are merged based on the modularity maximization. The performance of the proposed algorithm is studied through the simulation on the popular networks which indicates that the superiority of the proposed algorithm in comparison with well-known overlapping community detection methods.

The rest of the paper organized as follows: in the second section, we introduce iterative and stochastic algorithms which produce different results in each run. Evaluation of community detection algorithms are explained in section 3. In section 4, a new algorithm for overlapping community detection has been proposed. Section 5 gives a description of simulations and tests on the proposed community detection algorithms using the real social networks. Finally, section 6 concludes the paper.

II. MATERIAL AND METHODS

A. *Louvain algorithm*

Some of the algorithms do not produce deterministic results for each run. One of the most popular ones which was mentioned in the previous section is the Louvain algorithm. This method not only produces accurate results but also is one of the fastest algorithms in community detection. It can analyze a network made up of more than six million users only in less than a minute. This method which is based on the modularity maximization and it works in two phases. At the first phase, it tries to find the smallest communities according to maximizing the modularity. Then in the second phase, it builds a new network which each node is a community from the first phase. Iteratively, the Louvain accomplishes two phases to obtain the maximum value of modularity.

B. *Iterative algorithms*

Another category of algorithms are iterative algorithms such as genetic algorithms [16] and reinforcement learning algorithms [17]. This type of algorithms are also trying to search the problem space to find the best solution. In each iteration, according to an objective function (e.g., select at random with a given probability), a structure of the network is discovered. Measuring the quality of discovered structures for an algorithm based on reinforcement learning is a feedback from environment (as shown as Fig. 2).

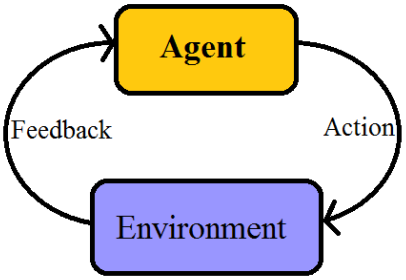


Figure 2. Feedback from environment in reinforcement learning algorithms

This feedback shows whether the last action of the algorithm was effective or not, then algorithm tries to improve its actions and will change the communities members to get a better result. This procedure will have continued till it converges to an optimal state. Although these methods are time consuming, for some reasons may provide the best accuracy.

III. Evaluation Methods

The key aspect of community detection is evaluation of the results of clustering methods. To evaluate the quality of clustering methods some measures are presented such as ARI, NMI, Pareto and Modularity. The well-known measure is modularity which is usually used for evaluating the results of community detection in real world networks and proposed by Newman. As given in Equation (1), where A is the adjacency matrix of the given network, m is the number of edges, $P_{i,j}$ is the probability of being an edge between node i and node j. V is the node set, and $\delta(c_i, c_j)$ is the function that returns 1 if nodes i and j belong to the same community, 0 otherwise. In equation (2), k_j^{in} is in-degree of node j and k_i^{out} is out-degree of node i. In this method the goal is to maximize density of communities in comparison to random graph density.

$$\boldsymbol{Q_d} = \sum_{i,j \epsilon V} \left[\frac{\boldsymbol{A_{i,j}}}{\boldsymbol{m}} - \frac{\boldsymbol{P_{i,j}}}{\boldsymbol{m}}\right] \boldsymbol{\delta(c_i, c_j)} \quad (1)$$

$$\boldsymbol{P_{i,j}} = \frac{\boldsymbol{k_i^{out} k_j^{in}}}{\boldsymbol{m}} \quad (2)$$

However, Q_d is common in disjoint methods; it is not working well for overlapping community detection algorithms. Accordingly, *Nicosia* et al. [18] proposed an extension of Newman modularity which is now common in community detection and called it Q_{ov} as given below

$$\boldsymbol{0 \leq \alpha_{i,c} \leq 1 \; \forall_i \epsilon V \, , \forall_c \epsilon C} \quad (3)$$

$$\boldsymbol{\sum_{c=1}^{|C|} \alpha_{i,c} = 1} \quad (4)$$

where $\alpha_{i,c}$ determines how node i belongs to community c.

$$\boldsymbol{\beta_{l,c} = F(\alpha_{i,c}, \alpha_{j,c})} \quad (5)$$

Function F in (5), is combination of belonging value of nodes i and j to community c. As an example, multiply of their values is $F(\alpha_{i,c}, \alpha_{j,c}) = \alpha_{i,c} * \alpha_{j,c}$

$$\boldsymbol{Q_{ov} = \frac{1}{m} \sum_{c \in C} \sum_{i,j \in V} [A_{ij} * r_{ijc} - \frac{k_i k_j}{m} s_{ijc}]} \quad (6)$$

$$\boldsymbol{r_{ijc} = \beta_{l,c} = F(\alpha_{i,c}, \alpha_{j,c})} \quad (7)$$

$$\boldsymbol{s_{ijc} = \frac{\sum_{i \in V} F(\alpha_{i,c}, \alpha_{j,c})}{|V|} * \frac{\sum_{j \in V} F(\alpha_{i,c}, \alpha_{j,c})}{|V|}} \quad (8)$$

$$\boldsymbol{\beta_{l(i,j),c} = \frac{1}{(1+e^{-f(\alpha_{i,c})})(1+e^{-f(\alpha_{j,c})})}} \quad (9)$$

Finally, equation (9) will be used for function F and after substituting in the mentioned equations, Q_{ov} will be equivalent to equation (10).

$$\boldsymbol{Q_{ov} = \frac{1}{m} \sum_{c \epsilon C} \sum_{i,j \epsilon V} [A_{i,j} \beta_{l(i,j),c} - P_{i,j} \beta_{l(i,j),c}{}^{in} \beta_{l(i,j),c}{}^{out}]} \quad (10)$$

IV. Proposed Algorithm

Our algorithm is based on the algorithms which described in section 2. This algorithm works in three phases. At the first step, the Louvain idea is used to have a disjoint community detection algorithm. The Louvain method not only can process on large scale social networks include of one hundred million users or one billion connection but also process a network include of one million nodes in less than a minute (~45sec) and a network with two million nodes in two minutes. Although, the Louvain have acceptable accuracy in detecting disjoint communities, it can't discover overlapping communities. Forasmuch, as the Louvain works greedy and in each step is trying to increase the modularity, so the results are not the same for each iteration. As we can see in Fig. 3, in each run it will return different output, especially in networks with overlapping communities. Shared nodes in each run will be joining to other community. Thus, at the first phase, after some iteration of running the Louvain method and recording the results, we have an expectation of belonging coefficient of each node to communities.

In the second phase, we have a belonging matrix with N row (number of nodes) and C column (number of communities). Based on the results of the first phase which is a disjoint structure recovery and according to the number of edges of each node which belongs to the community c_k, matrix elements in column k will be being updated.

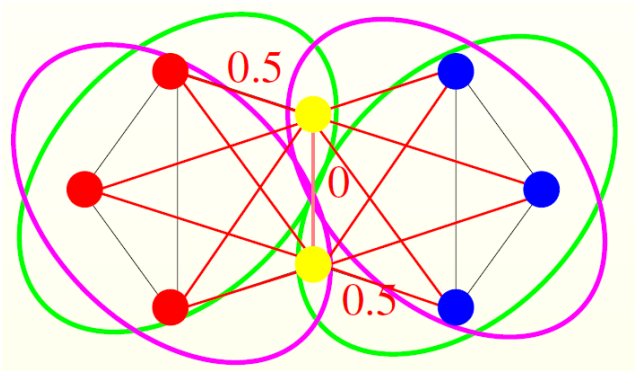
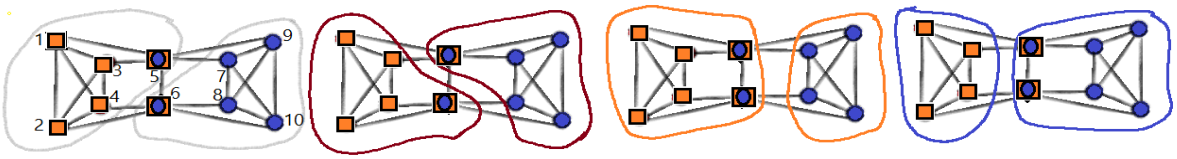


Figure 3. Running several times of the Louvain algorithm on the same given network[19]



node	C_1	C_2
1	**4**	**0**
2	**3**	**1**
3	**4**	**0**
4	**3**	**1**
5	**2**	**3**
6	**3**	**2**
7	**1**	**3**
8	**0**	**4**
9	**1**	**3**
10	**0**	**4**

a) First run

node	C_1	C_2
1	7	1
2	7	1
3	7	1
4	7	1
5	5	5
6	5	5
7	1	7
8	1	7
9	1	7
10	1	7

b) Second run

node	C_1	C_2
1	11	1
2	11	1
3	11	1
4	11	1
5	8	7
6	8	7
7	2	10
8	2	10
9	2	10
10	2	10

c) Third run

node	C_1	C_2
1	14	2
2	14	2
3	14	2
4	14	2
5	10	10
6	10	10
7	2	14
8	2	14
9	2	14
10	2	14

d) Fourth run

Figure 4. How algorithm works

To give a sense of this operation, in Fig. 4 in second run, node 6 is member of community 1 but have two edges in community 1 and three edges in community 2, then matrix element value in row 6 of column 1 will be updated from 3 to 5 and in row 6 of column 2 from 2 to 5. At first, all matrix elements are zero but they are increasing during the algorithm is running. At last, all values will be normalized in range of [0 1] and the values indicate belonging coefficient of each node to each community. If the belonging coefficient of node i to community k is higher than threshold Lambda, then node i belongs to community k.

$$\boldsymbol{\lambda = \frac{1}{ov+1}} \qquad (11)$$

In equation (11), the parameter ov could be set by user. This parameter shows the maximum number of communities which nodes could be joined to. Fig. 4 shows how this phase is working. Also, Table 1 depicts the belonging matrix of clustering of Fig. 4 after normalization.

TABLE 1. BELONGING MATRIX OF FIG. 4 AFTER NORMALIZATION

node	C_1	C_2	node	C_1	C_2
1	0.88	0.12	6	0.50	0.50
2	0.88	0.12	7	0.12	0.88
3	0.88	0.12	8	0.12	0.88
4	0.88	0.12	9	0.12	0.88
5	0.50	0.50	10	0.12	0.88

$ov = 2$, $\lambda = 0.33$

For initialization, each node is a community, for example for a network with N nodes, the size of belonging matrix will be $N \times N$. That is for initialization, i^{th} node is i^{th} community and we have N communities. As the algorithm proceeds, many of columns of belonging matrix will be omitted. In phase three, as some of communities can possibly share big part of their substructures, under the condition of increasing Q_{ov} modularity, we need to merge them if they are highly overlapped or very close to each other. Testing all possible cases of combinations is impossible or taking long time. To achieve this goal, in this phase, we use a greedy approach. To proceed, at each step, we select a community and try to find the first community which in combination eventuated a higher modularity value.

To illustrate this method, let's consider an example.

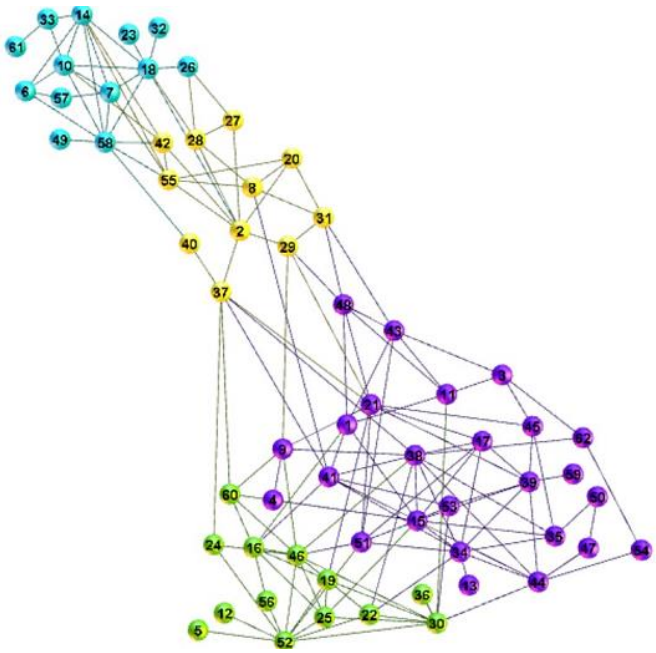


Figure 5. Output of running first phase of algorithm on Dolphin dataset

Fig. 5 is the output of one iteration for the first phase of algorithm after running on Dolphins network dataset. As we can see, graph nodes was clustered and discovered disjoint communities. Some nodes like nodes 29, 30, 31 and 37 have equal number of edges in two or three communities. So we can say they belongs to more than one community.

Fig. 6 shows that each node can belong to more than one community. If the belonging coefficient of each node to each

community is higher than the given threshold, it belongs to that community. For instance, based on the belonging matrix, belonging coefficient of node 31 to blue community is 0.6 and to orange community is 0.4. Then with threshold value λ<= 0.4 node 31 belongs to both communities blue and orange.

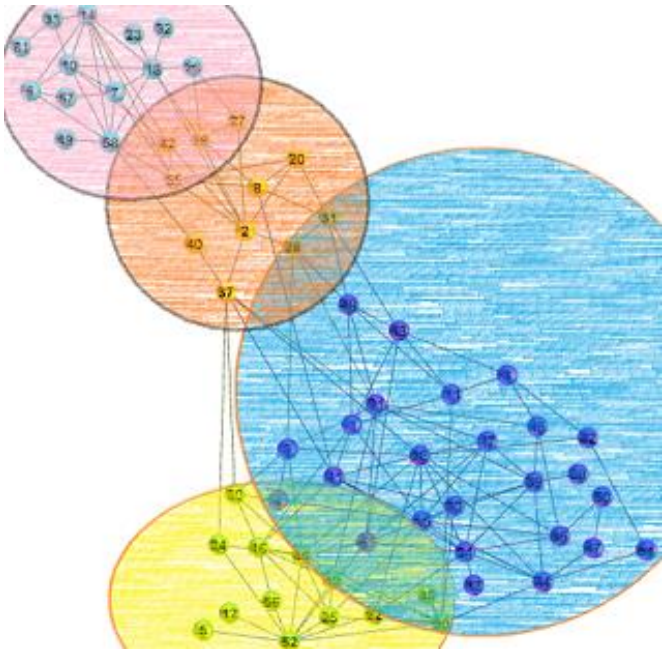


Figure 6.Output of second phase of algorithm

Fig. 7 is output of third phase of algorithm. The two communities can be merged as one, if merging resulted higher modularity. In this example, blue and yellow communities are merged together. Similarly, communities orange and pink are merge together.

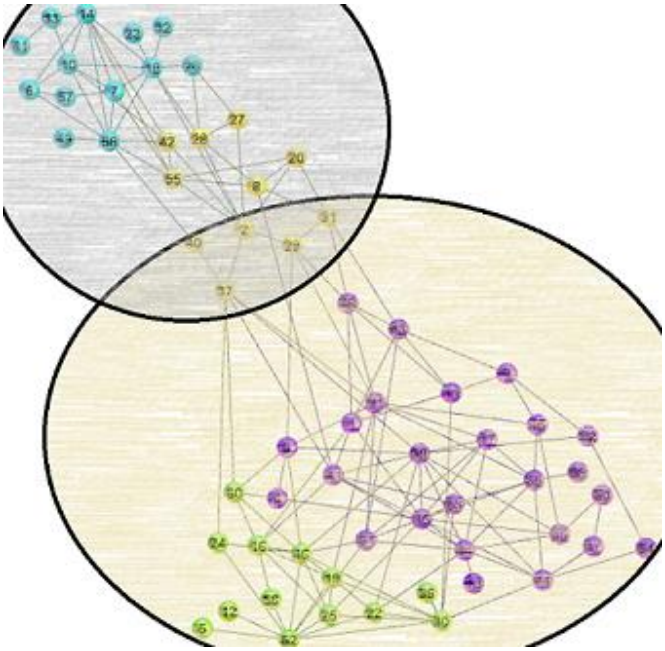


Figure 7.Output of second phase of algorithm

The time complexity of the proposed algorithm could be calculated separately in three phases. At the first phase, it seems that the time complexity of the Louvain method is $O(n\log n)$. At the second phase, k ($k << n$) operations for overlapped nodes are performed by updating the belonging matrix. The time complexity of the last phase consists of n^3 operations for computing the Nicosia modularity. Since, three phases are performed sequentially, hence the total time complexity of the algorithm is $O(n\log n) + O(k) + O(n^3) = O(n^3)$. It is worth to note that the impact of the role of the third phase on the quality of results is investigated. However, last phase helps us to improve the quality of modularity, as we can see in Fig. 8 improvement is legible, hence in some circumstances that time is more valuable last phase can be ignorable.

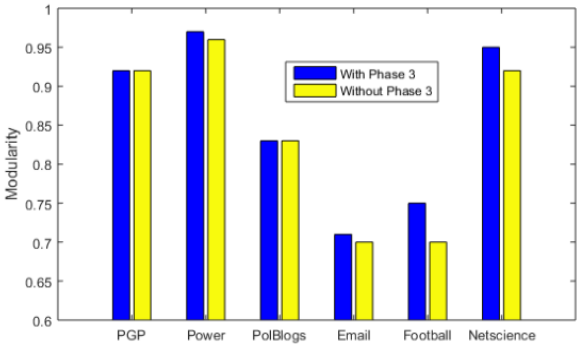


Figure 8.Comparing modularity with & without phase 3

V. EXPERIMENTAL RESULTS

This section describes the results of "belonging matrix" method. The performance of our method have been compared to other algorithms such as COPRA [13], SLPA [14], CONGA [11], and GA-Net [15]. To evaluate our algorithm an accurate measurement like Nicosia Modularity (Q_{ov}) was used. We use six well-known real world datasets which described in table 2. In Table 3 we give the experimental results of the most efficient algorithms in overlapping community detection and our method. The outcome shows us a significant superiority for our method among other algorithms. Results show a clear correlation between algorithms outcomes and their parameters. Results of COPRA are tightly depending on the parameter v adjustment, and the SLPA's dependency on the parameter r. In CONGA we have to know the number of communities, and in C-Finder we need to know the size of cliques. In contrast, our belonging matrix method only has parameter Lambda, after examining, we experimentally found the best value of this parameter which is working in all datasets is 0.368. Then, in our method we do not need to adjust any parameter, also there is no need to have any information about the network.

TABLE 2. DESCRIPTION OF DATASETS FOR EXPERIMENTS

Network	Avg. node degree	Edges	Nodes
PGP	4.5	24316	10680
Power	2.6	6594	4941
PolBlogs	25.5	19090	1490
Email	9.6	10902	1133
Football	10.6	613	115
Netscience	4.8	1136	379

TABLE 3. Q_{ov} OF OUR METHOD IN COMPARISON WITH OTHER ALGORITHMS ON REAL SOCIAL NETWORKS

Network	Belonging Matrix	GA-net	C-finder	CONGA	COPRA	SLPA
PGP	**0.92**	0.80	0.57	0.86	0.80	0.83
Power	**0.97**	-	-	**0.97**	0.41	-
PolBlogs	**0.83**	0.41	0.40	0.57	0.80	-
Email	**0.71**	0.39	0.46	0.63	**0.71**	0.67
Football	**0.75**	0.60	0.64	0.64	**0.75**	0.71
Netscience	**0.95**	0.55	0.61	0.91	0.87	0.86

VI. CONCLUSION AND FUTURE WORKS

To sum up, belonging matrix method is able to bind to any disjoint clustering algorithms which are working iteratively and algorithms which do not have deterministic results in output. Our method can be applied for both disjoint and overlapping communities. In addition, the proposed method does not have any sensitive parameter for adjustment, this is in contrast to other algorithms that need to adjust parameters accurately and also some of them need to know the number of clusters in prior. However, knowing the exact number of clusters will help to optimize the speed of clustering and consequently last phase of the algorithm can be omitted. Finding a faster method for the last phase of the algorithm will be a goal as possible future works.

REFERENCES

[1] M. M. D. Khomami, A. Rezvanian, and M. R. Meybodi, "Distributed learning automata-based algorithm for community detection in complex networks," *Int. J. Mod. Phys. B*, vol. 30, no. 8, p. 1650042, 2016.

[2] K. Deb, "Multi-Objective Optimization," in *Search methodologies*, Springer, 2014, pp. 403–449.

[3] P. Barbillon, S. Donnet, E. Lazega, and A. Bar-Hen, "Stochastic Block Models for Multiplex Networks: An Application to Networks of Researchers," *ArXiv150106444 Stat*, Jan. 2015.

[4] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato, "Finding Statistically Significant Communities in Networks," *PloS One*, vol. 6, no. 4, p. e18961, 2011.

[5] K.-J. Hsiao, K. Xu, J. Calder, and A. O. Hero, "Multi-Criteria Anomaly Detection Using Pareto Depth Analysis," in *Advances in Neural Information Processing Systems*, 2012, pp. 845–853.

[6] M. Hoffman, D. Steinley, and M. J. Brusco, "A Note on Using the Adjusted Rand Index for Link Prediction in Networks," *Soc. Netw.*, vol. 42, pp. 72–79, Jul. 2015.

[7] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New J. Phys.*, vol. 11, no. 3, p. 033015, 2009.

[8] M. E. Newman and M. Girvan, "Finding and Evaluating Community Structure in Networks," *Phys. Rev. E*, vol. 69, no. 2, p. 026113, 2004.

[9] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech. Theory Exp.*, vol. 2008, no. 10, p. P10008, 2008.

[10] S. Maity and S. K. Rath, "Extended Clique percolation method to detect overlapping community structure," in *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI*, 2014, pp. 31–37.

[11] S. Gregory, "An Algorithm to Find Overlapping Community Structure in Networks," in *Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases*, Berlin, Heidelberg, 2007, pp. 91–102.

[12] S. Gregory, "A Fast Algorithm to Find Overlapping Communities in Networks," in *Machine Learning and Knowledge Discovery in Databases*, W. Daelemans, B. Goethals, and K. Morik, Eds. Springer Berlin Heidelberg, 2008, pp. 408–423.

[13] S. Gregory, "Finding overlapping communities in networks by label propagation," *New J. Phys.*, vol. 12, no. 10, p. 103018, 2010.

[14] J. Xie, B. K. Szymanski, and X. Liu, "SLPA: Uncovering Overlapping Communities in Social Networks via a Speaker-Listener Interaction Dynamic Process," in *2011 IEEE 11th International Conference on Data Mining Workshops*, 2011, pp. 344–349.

[15] C. Pizzuti, "GA-Net: A Genetic Algorithm for Community Detection in Social Networks," in *Parallel Problem Solving from Nature – PPSN X*, G. Rudolph, T. Jansen, N. Beume, S. Lucas, and C. Poloni, Eds. Springer Berlin Heidelberg, 2008, pp. 1081–1090.

[16] Y. Atay and H. Kodaz, "A New Adaptive Genetic Algorithm for Community Structure Detection," in *Intelligent and Evolutionary Systems*, K. Lavangnananda, S. Phon-Amnuaisuk, W. Engchuan, and J. H. Chan, Eds. Springer International Publishing, 2016, pp. 43–55.

[17] Y. Zhao, W. Jiang, S. Li, Y. Ma, G. Su, and X. Lin, "A cellular learning automata based algorithm for detecting community structure in complex networks," *Neurocomputing*, vol. 151, Part 3, pp. 1216–1226, Mar. 2015.

[18] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri, "Extending the definition of modularity to directed graphs with overlapping communities," *J. Stat. Mech. Theory Exp.*, vol. 2009, no. 03, p. P03024, 2009.

[19] Q. Wang, "Overlapping community detection in dynamic networks," May-2012. [Online]. Available: https://tel.archives-ouvertes.fr/tel-00701217/document. [Accessed: 01-Jul-2016].