2020 8th Iranian Joint Congress on Fuzzy and intelligent Systems (CFIS), September 2-4, 2020, Mashhad, Iran

SIG-CLA: A Significant Community Detection based on Cellular Learning Automata

Mohammad Mehdi Daliri Khomami [1] Alireza Rezvanian [2,3] Ali Mohammad Saghiri [4] Mohammad Reza Meybodi [5]

[1]Soft computing laboratory, Department of Computer Engineering, Amirkabir University of Technology, M.daliri@aut.ac.ir
[2]Department of Computer Engineering, University of Science and Culture, Tehran, Iran, rezvanian@usc.ac.ir
[3]School of Computer Science, Institute for Research in Fundamental Sciences (IPM), Tehran, Iran,
[4]Soft computing lab, Computer Engineering Department, Amirkabir University of Technology, saghiri@aut.ac.ir
[5]Soft computing lab, Computer Engineering Department, Amirkabir University of Technology, mmeybodi@aut.ac.ir

Abstract— **Detecting community, as the fundamental task for the study of the network, reveals a hopeful approach to investigating the functional and topological properties of real networks. Recently, several algorithms for detecting community have been introduced from various perspectives. A typical algorithm which is noticed by many researchers in this scope is Modularity optimization. These algorithms significantly restricted in resolution limits that they may miss detecting communities that are less than a particular size. The paper presents an algorithm with the aid of ICLA (irregular cellular learning automata) for the detection of community structures (called SIG-CLA) in social networks. In the SIG-CLA, the ICLA is formed by the input network, and the communities are detected by communicating with both the local environment and the global environment via the significant function in the ICLA. Promising experimental results are presented to confirm the advantages of SIG-CLA with respect to Modularity and NMI.**

Keywords— Community Discovery, Social Network, Learning Automata, Irregular Cellular Learning Automaton.

I. INTRODUCTION

In numerous practical applications in different domains, the network or graphs modeling, which composed as nodes, and relations as edges are conventional. In the network theory, communities are an essential class of sub-network that defied by the group of highly linked vertices and weakly linked with other parts of the network. Due to various applications of detecting community structures, it is a basic issue in the social network analysis. Detecting communities are connected with functions of the networks under study; hence, identifying this functionality provides perception into the organization of the graph. Moreover, detecting community in a graph is known as an NP-hard problem and computationally challenging task [1].

Many scholars have been proposed community detection algorithms since the notable reported is presented by Girvan and Newman [2]. A widespread class of algorithms for finding communities is the objective function based in which tries to optimize the modularity [3],[4]. Also, an example of other techniques based on machine learning approaches is node2vec [5]. Besides, the Bayesian method [9] and artificial neural network technique [8], were used for community discovery.

Modularity has been used as a measurement that studies the importance of structures of the community in the graph. This measurement is presented by Newman et al. [7]. One of formalism for detecting community is optimizing modularity function, which has been proved to be NP-hard. However, *Fortunato et al.* were pointed out that modularity optimization may not lead to proper results when the size of communities is small [10]. In this manner, several communities may be grouped as one and are referred to as a resolution limit for modularity. To treat with the problem of resolution limit in detecting community, different functions have been suggested by many researchers to improve the problem related to the resolution limit. [11]–[14].

In general, community detection can be classified into the domain of a clustering. A fundamental notion in clustering is utilizing similarities which consider the neighborhood to detect communities. However, this local structure misses the global structures, such as the size of communities in general for detecting communities. Furthermore, the communities which produced just by the information of global structure, ignore the details of communications, may decrease the effectiveness of community detection algorithms [15]–[17], [33-34].

The CLA (cellular learning automaton) [7] is an amalgam model of LA (learning automata) and CA (cellular automata) [22]. CLA is a cellular automaton, where an LA is placed in every cell. CLA capability is increased to cellular automata due to the LA exists in each cell by providing the ability of learning. It is also more powerful to single LA, due to residing LA in different cells can cooperate to create a complex structure. Recently, many applications are used CLA in various domains, such as evolutionary computing [18], P2P networks [19], and mobile networks [20].

To adders the above issue, this paper proposed an algorithm based on CLA called SIG-CLA for detecting community in graphs. Briefly, SIG-CLA first considers the whole network as an isomorphic to an ICLA. Second, using the learning process of CLA and by cooperation among both the local environment and global environment, a candidate solution is constructed, and SIG-CLA evolves and gradually reveals communities.

The article is organized as follows: In section 2, ICLA and its background are briefly expressed. In section 3, the structure of the suggested (SIG-CLA) is presented. In section 4, by the use of different experiments, the effectiveness of the SIG-CLA on both the artificial and real networks is shown compared with other classical detecting community methods. Finally, section 5 gives the concluding remarks.

II. AUTOMATA THEORY

In this section, we first briefly explain the theory of CA and LA. Then, Irregular CLA (ICLA) as a new variant of CLA is introduced.

A. Cellular Automata

Cellular Automata models are interesting models in mathematics [21]. Theses model consists of a set of cells and also a structure for organizing the cells. In these models, a number of states are defined for the cells. Every cell selects a state of the other states. In each cell, the operation of state selection is performed locally. During this operation, each cell makes a decision about its states base on its local observation, which is the states of its neighboring cells.

B. Learning Automata

The theory of learning automata belongs to the primary models of reinforcement learning [22]. These model can be considered as an agent which operate in unknown stochastic environments. In these models, a set of actions is defined for the agent and also a set of feedback that is defined to represent the response of the environment to the actions of the agents. The agent using learning automata theory tries to find an appropriate action. The appropriate action is one of the actions from an action set that triggers the environment to generate reward signals in an asymptotical manner. Figure 1 shows the relationship between LA and its environment.

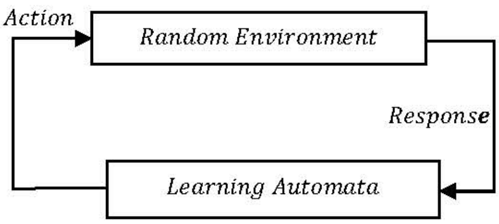

Figure 1. The relationship between the LA and its random environment

The structure of the learning automata used in this paper may be defined with 4-tuple $\{\alpha, \beta, P, T\}$ such that:

- $\alpha = \{\alpha_1, \dots, \alpha_r\}$ is the limited number of actions
- $\beta = \{\beta_1, \dots, \beta_m\}$ denotes the reinforcement signal
- $P = \{p_1, \dots, p_r\}$ is the probability vector

$T[\alpha(n), \beta(n), p(n)]$ is learning algorithm. Formula (1) and (2) show update in vector of probability action in each iteration. Learning algorithm moderates the vector of probability of action using formula (1) for promising responses as:

$$p_i(t+1) = p_i(t) + a[1 - p_i(t)] \qquad (1)$$
$$p_j(t+1) = (1-a)p_j(t)$$

and formula (2) for unfavorable ones:

$$p_i(t+1) = (1-a)p_i(t)$$
$$p_j(t+1) = \left(\frac{b}{t-1}\right) + (1-b)p_j(t) \qquad (2)$$

The learning rate of the LA can be managed by two parameters a and b, where a denotes the reward and b denotes the penalty parameters [23]–[25].

C. Cellular learning automata

The theory of CLA brings together the learning capabilities of LA and the distributed computation characteristics of the CA. The core operation of the learning algorithm in CLA can be expressed as follows: A d-dimensional CLA is a structure $CLA = (Z^d, \Phi, A, N, F)$, where

1. $Z^d$is a matrix of d-tuples.
2. Φ is a limited number of states.
3. A is the number of learning automata, where every LA is placed the corresponding cell of cellular automata.
4. $N = \{x_1, \dots, x_m\}$ is a limited number of of Z^d called neighboring hood vector, where $x_i \in Z^d$ of the CA.
5. $F: \Phi_m \to \beta$ is the local rule of the CLA, in which β denotes the values of reinforcement.

D. Irregular Cellular Learning Automata

The Irregular CLA (ICLA) is a new version of CLA, where the limitation on the regular structure in conventional CLA is eliminated. Many problems cannot be modeled by a grid structure.

III. PRELIMINARIES AND PROBLEM STATEMENTS

A network system may be considered as a graph $G = < N, E >$, in which N indicates nodes and shows the users, and E represents the edges or relationship in the networks. Let $SF(N)$ indicates a significant quality function, which designates ratio of quality to any partition P of N. In Graph G, community detection aims to find k partitions $P = \{C_1, \dots, C_k\}$ of the set V of vertices so that $SF(P)$ is optimal. We note that no assumption is made regarding the number or scale of the communities. In the subsequent, we explain the step of the algorithms. Moreover, we call our algorithm as SIG-CLA in brief for simplicity.
The SIG-CLA consists of 3 main steps including: *Initialization Phase*, *Solution Description Phase* and *Objective function evaluation.*

- ***Initialization Phase***

Initialization is one of primary phase which is done once in the CIG-CLA. In this phase, an ICLA is mapped to the network. In the mapping, a one to one function operates to assigning each node to each cell. The cell's state of an LA is determined by current action which is taken by the LA. Each learning automata LA_i residing in vertex v_i may be defined by a triple (α, β, p), such that:

- $\alpha_i = \{\alpha_{i1}, \ldots, \alpha_{ir}\} = N(i)$ indicates the set of action and $N(i)$ shows that neighborhood set i in the network.
- $\beta_i = \{0,1\}$ Indicates reward and penalty value of LA_i which is taken from the environment.
- $P_i = (p_{i1}, p_{i2}, \ldots, p_{ir})$ shows the action probability vector in which p_{ij} is the probability of selecting action α_{ij} by learning automata LA_i.

With the aid of the cells state, a solution vector S is constructed. By the use of S, it is possible to determine the current actions selected by all the LA.

Therefore, during iteration t, the solution can be described by equation (3).

$$S(t) = \{\alpha_1(t), \alpha_2(t), \ldots, \alpha_n(t)\} \qquad (3)$$

Where $\alpha_i(t)$ indicates the selected action by learning automata LA_i at iteration t. The mentioned procedure updates the solution vector as the algorithm grows.

- ***Solution Description Phase***

In this phase of SIG-CLA, a membership function is applied to reveal the communities. In SIG-CLA, with the aid of this function, the cardinality number of communities is obtained. Therefore it is not proper for serving members of communities. We utilize the locus-based adjacency representation presented in [26]. The locus-based representation contains vector $S = (s_1, s_2, \ldots, s_n)$. In this vector for each $s_i \in S$ denotes that vertex v_i and vertex s_i be in own community. To obtain more information about solution vector please refer to [26].

Since the solution vector contains the edges; therefore, a transform mechanism by BFS or a DFS algorithm is done to get communities in the network.

- **Objective Function evaluation**

One of the capabilities of open synchronous CLA (OSCLA) is working with complicated patterns like communities. In OSCLA, an LA has two different environments, including local and global. The local environment is limited to each LA separately, while the global environment includes all the LA collectively. Therefore, two local and global are applied for the evaluation of the communities revealed by SIG-CLA. In this paper, the significance criteria are used as an objective function for the global environment. The significance function indicates the probability of finding a subgraph including n_c nodes with density q from a random graph with n node and density p is equal by the following equation:

$$Pr\big(S(n_c, q) \subseteq \mathcal{G}(n, p)\big) = e^{\theta(-\binom{n_c}{2}D(q||p))} \qquad (4)$$

where $D(q||p)$ is the distance from p and q which is computed by equation (5):

$$D(q \parallel p) = (1-q)\,log\frac{1-q}{1-p} + q\ log\frac{q}{p} \qquad (5)$$

We note that, the equation (4) is proposed for a community of network and by extending the formula (4) for all network we obtain equation (6):

$$Pr(\sigma) = \prod_c exp(-\binom{n_c}{2}D(p_c \parallel p)) \qquad (6)$$

Therefore, signidicant is obtained by equation (7):

$$S(\sigma) = -log\,Pr(\sigma) = \sum_c \binom{n_c}{2} D(p_c \parallel p) \qquad (7)$$

IV. PROPOSED ALGORITHM

The SIG-CLA works as follows. In each running of the SIG-CLA algorithm, each different LA's selects an action simultaneously. The selected actions by LA's makes to generate a candidate solution vector $S(t)$. Afterwhile, by decoding and conversion, the solution vector changed into the membership vector $C(t)$. Using both local and global environments, the significant function $SF(t)$ and density of nodes inside communities are determined. Based on two following conditions which are obtained by both local and global the SIG-CLA decide to guide LAs by reward or penalize.

1- The $SF(t)$ current detected communities are not less than the maximum value of significant which is gained up to iteration t.

2- For nodes in each community must be adjacent to more than half of vertices that are present in their communities.

If any of the conditions are not satisfied, then LA's would get penalize from the both environments. Then, during the running SIG-CLA, the optimal action for finding communities is determined.

The SIG-CLA is proceeding in some consecutive cycles until the community structure remains unchanged.

The flowchart of the SIG-CLA for detecting community is shown in figure 2.

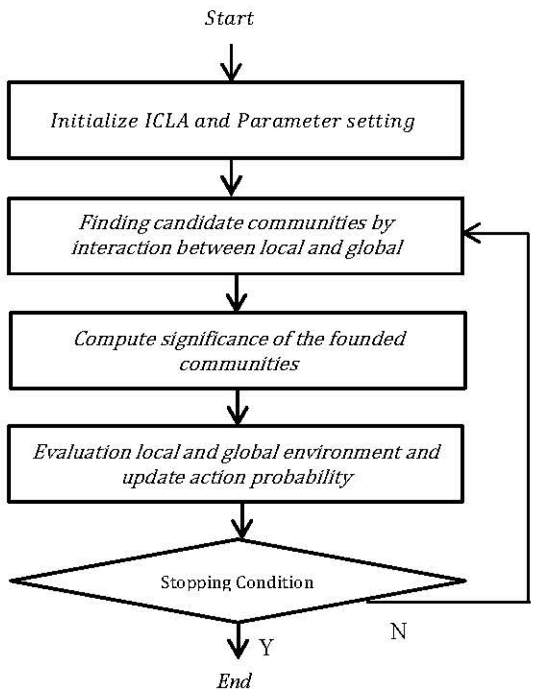

Figure 2. Flowchart of the SIG-CLA

V. Experiential Simulations

To study the efficiency of the SIG-CLA, we carried out some simulations on the popular artificial network called LFR [27]. We have used this synthetic network because of its property for the size of community and degree of node follows power-law degree distribution with γ and β exponents respectively.

In LFR $\boldsymbol{M}$ is mixing parameter, which means that for each vertex, a fraction, $1 - \boldsymbol{M}$ of links is connected with the same group of nodes and $\boldsymbol{M}$ of its links is connected with nodes of different groups. Table 1 shows the basics of LFR networks that are used for experiments. Besides, the SIG-CLA is compared with several well-known algorithms such as CNM [4], MOGA [28], GA-NET [29], MOCD [30], Meme.net [31] and CLA-net [32] in term of Normalized Mutual Information (NMI).

Table 1: Configuration of the LFR benchmark networks

Parameter	Values
Size of vertices $\boldsymbol{N}$	1,000
Avg. degree $\boldsymbol{K}$	5-20
Maximum community size ($\boldsymbol{Max_c}$)	50
Minimum community size ($\boldsymbol{Min_c}$)	20
Mixing parameter ($\boldsymbol{\mu}$)	0-0.8

A. Normlized Mutal Information

The quality of an algorithm for community detection on a known network can be computed by NMI (Normalized Mutual Information as given in equation (9).

$$NMI(P,Q) = \frac{-2\sum_{p\in P}\sum_{q\in Q}|p\cap q|.\log(\frac{|p\cap q|n}{|p||q|})}{\sum_{p\in P}|p|log\left(\frac{|p|}{n}\right)+\sum_{q\in Q}|q|.log\left(\frac{|q|}{n}\right)} \quad (9)$$

where P is the known community set and Q is the found community set by the algorithm. The NMI value is between $[0,1]$, where the NMI value nearby 1 is preferable [6]. The NMI is useful for known networks.

B. Experiments

- **Experiment I**

In this section, this test is conducted to investigate the efficiency of the SIG-CLA in assessment with famous algorithms for detecting community including MOGA [27], GA-NET [28], CNM [3], MOCD [29], Meme.net [30] and CLA-net [31] in terms of NMI depicted in Figure 3. In order to the fairness and correctness for comparison of the algorithms, we have reported 30 different independent runs of the proposed algorithm in the experiment.

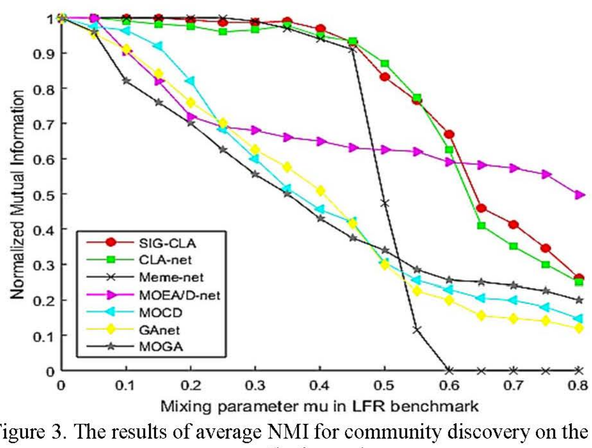

Figure 3. The results of average NMI for community discovery on the LFR synthetic graphs.

From the result, we may conclude that when $\mu<0.1$, there is no difference in the algorithm. However, by increasing the mixing parameter, the efficiency of the algorithms except for SIG-CLA, CLA-net, MIGA, and Meme-net are rigorously decreased and unable to reveal accurate community structure. We note that, when $\mu<0.4$ the SIG-CLA finds exact communities of the network. In addition other algorithms such as CLA-net, MIGA, and Meme-net give proper result. For $0.4<\mu<0.65$ SIG-CLA expect $\mu=0.5$ is superior with respect to other algorithms. But $\mu>0.65$ the MOEA/D-net is superior. It is notable that, for $\mu<0.65$ the efficiency of the MOEA/D-net is low which is far from the structure of the real networks. Finally, we can expressed the SIG-CLA is outperforms from other algorithms in terms on NMI.

- **Experiment II**

In this experiment, we have selected the CLA-net algorithm from other algorithms due to better performance. For this purpose, we have plotted NMI versus different mixing parameter, which is varying from 0 to 0.5 with 0.05 step length. The lower, upper, and average result is demonstrated in figure 4.

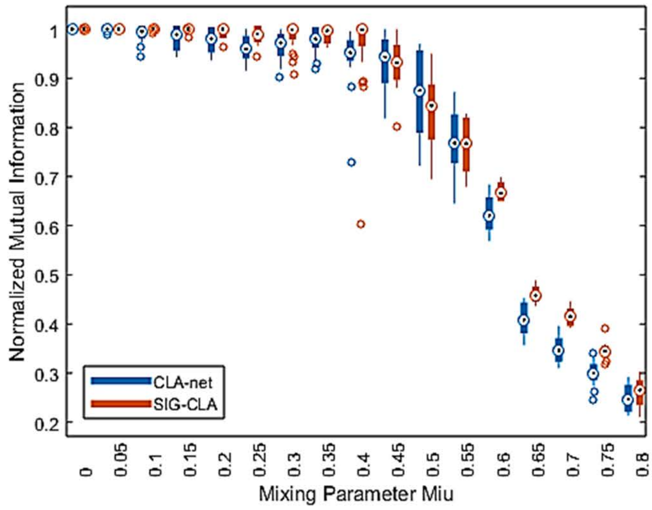

Figure 4. Comparison between SIG-CLA and CLA-net in terms of the distribution results.

As can be shown from the result, both CLA-net and SIG-CLA, when the scattering of the results is close to 0.5, is high. This phenomenon is happening when and out-degree and in-degree of communities are not notable. Hence, by executing the two algorithms independently, we get different solutions. But, totally the SIG-CLA shows better performance than CLA-net due to the variance of the obtained results is less than the CLA-net.

- **Experiment III**

In this experiment, we are trying to verify the superiorities of the SIG-CLA from the test of statistical. The statistical results of counterpart algorithms with the SIG-CLA algorithm and CLA-net by the t-Test as two-tailed and 58 degrees of freedom with a confidence level of significance at 0.95 are given in Table 2. In Table 2, the t-Test comparing algorithms: SIG-CLA vs. CLA-net when SIG-CLA is significantly better is shown as highlighted text. To do this experiment, we compute the t-test value versus different mixing parameters for the test network.

Table 1: Comparison SIG-CLA and CLA-net in terms of t-test

μ	CLA-net	SIG-CLA
0	1±0	1±0.0002
0.05	0.9997±0.0019	1±0
0.1	0.991±0.0122	0.9997±0.0014
0.15	0.9824±0.0194	0.9994±0.0031
0.2	0.9766±0.0215	0.994±0.0085
0.25	0.9598±0.0214	0.9878±0.0132
0.3	0.966±0.0252	0.9871±0.0236
0.35	0.9776±0.0212	0.9896±0.013
0.4	0.9484±0.0473	0.9681±0.0768
0.45	0.9337±0.0488	9305±0.0399
0.5	0.8711±0.0779	0.8313±0.0736
0.55	0.7739±0.0589	0.7639±0.0478
0.6	0.625±0.0343	0.6713±0.0141
0.65	0.4079±0.0261	0.4603±0.012
0.7	0.3494±0.0219	0.414±0.0128
0.75	0.3002±0.0195	0.3446±0.0122
0.8	0.2493±0.0241	0.2621±0.0226

As can be seen in Table1, we may observe in 11 cases the SIG-CLA are superior, in 5 are the same, and in 1 case, the SIG-CLA is worse. Therefore, someone can conclude that the SIG-CLA overly all outperforms other algorithms.

VI. CONCLUSTION

In this paper, we proposed an algorithm based on CLA called SIG-CLA, which is applied for the detecting community problem. The SIG-CLA uses critical criteria to call Significant as a global measure to quantify community feasibility. In addition, a local density is considering by the SIG-CLA for each node to control the density of nodes within communities. These measures are applied as local and global environments in the algorithms. With the aid of the cooperation between both local and global environments and the learning mechanism, the SIG-CLA tries to find optimal communities iteratively. Also, this policy provides a new way to solve the resolution limit effects on social networks. The performance of the SIG-CLA was investigated by simulations and compared with the familiar algorithms for community discovery. The simulation results reveal the superiorities of the SIG-CLA with respect to NMI.

VII. REFERENCES

[1] S. Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3–5, pp. 75–174, 2010.

[2] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[3] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of statistical mechanics: theory and experiment*, vol. 2008, no. 10, p. P10008, 2008.

[4] A. Clauset, M. E. Newman, and C. Moore, "Finding community structure in very large networks," *Phys Rev E*, vol. 70, no. 6, p. 066111, 2004.

[5] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[6] M. M. D. Khomami, A. Rezvanian, M.R. Meybodi, A new cellular learning automata-based algorithm for community detection in complex social networks. Journal of computational science, 24, 413-426, 2018.

[7] A. Rezvanian, A. M. Saghiri, S. M. Vahidipour, M. Esnaashari, M. R. Meybodi. "Cellular Learning Automata." In Recent Advances in Learning Automata, pp. 21-88, 2018.

[8] J. C. Lv, K. K. Tan, Z. Yi, and S. Huang, "A family of fuzzy learning algorithms for robust principal component analysis neural networks," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 1, pp. 217–226, 2009.

[9] M. Mørup and M. N. Schmidt, "Bayesian community detection," *Neural computation*, vol. 24, no. 9, pp. 2434–2456, 2012.

[10] A. Lancichinetti and S. Fortunato, "Consensus clustering in complex networks," *Scientific reports*, vol. 2, p. 336, 2012.

[11] S. Zhang, R.-S. Wang, and X.-S. Zhang, "Identification of overlapping community structure in complex networks using fuzzy c-means clustering," *Physica A*, vol. 374, no. 1, pp. 483–490, 2007.

[12] Y. Pan, D.-H. Li, J.-G. Liu, and J.-Z. Liang, "Detecting community structure in complex networks via node similarity," *Physica A*, vol. 389, no. 14, pp. 2849–2857, 2010.

[13] K. R. Žalik, "Maximal neighbor similarity reveals real communities in networks," *Scientific reports*, vol. 5, p. 18374, 2015.

[14] S. Ahajjam, M. El Haddad, and H. Badir, "A new scalable leader-community detection approach for community detection in social networks," *Social Networks*, vol. 54, pp. 41–49, 2018.

[15] O. R. B. Speily, A. Rezvanian, A. Ghasemzadeh, A. M. Saghiri, S. M. Vahidipour, Lurkers Versus Posters: Investigation of the Participation Behaviors in Online Learning Communities. In Educational Networking (pp. 269-298). Springer, Cham, 2020.

[16] M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys Rev E*, vol. 74, no. 3, p. 036104, 2006.

[17] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical review E*, vol. 76, no. 3, p. 036106, 2007.

[18] R. Vafashoar and M. R. Meybodi, "Cellular learning automata based bare bones PSO with maximum likelihood rotated mutations," *Swarm and evolutionary computation*, vol. 44, pp. 680–694, 2019.

[19] A. M. Saghiri and M. R. Meybodi, "An adaptive super-peer selection algorithm considering peers capacity utilizing asynchronous dynamic cellular learning automata," *Applied Intelligence*, vol. 48, no. 2, pp. 271–299, 2018.

[20] M. R. Mirsaleh and M. R. Meybodi, "Assignment of cells to switches in cellular mobile network: a learning automata-based memetic algorithm," *Applied Intelligence*, vol. 48, no. 10, pp. 3231–3247, 2018.

[21] S. Wolfram, *Cellular automata and complexity: collected papers*. CRC Press, 2018.

[22] K. S. Narendra and M. A. Thathachar, *Learning automata: an introduction*. Courier Corporation, 2012.

[23] M. R. Mirsaleh and M. R. Meybodi, "A Michigan memetic algorithm for solving the community detection problem in complex network," *Neurocomputing*, vol. 214, pp. 535–545, 2016.

[24] R. Vafashoar and M. R. Meybodi, "Reinforcement learning in learning automata and cellular learning automata via multiple reinforcement signals," *Knowledge-Based Systems*, vol. 169, pp. 1–27, 2019.

[25] B. Moradabadi and M. R. Meybodi, "Wavefront cellular learning automata," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 28, no. 2, p. 021101, 2018.

[26] Y. Park and M. Song, "A genetic algorithm for clustering problems," in *Proceedings of the third annual conference on genetic programming*, 1998, vol. 1998, pp. 568–575.

[27] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Phys Rev E*, vol. 78, no. 4, p. 046110, 2008.

[28] C. Pizzuti, "A multiobjective genetic algorithm to find communities in complex networks," *IEEE T Evolut Comput*, vol. 16, no. 3, pp. 418–430, 2011.

[29] C. Pizzuti, "Ga-net: A genetic algorithm for community detection in social networks," in *International conference on parallel problem solving from nature*, 2008, pp. 1081–1090.

[30] C. Shi, Z. Yan, Y. Cai, and B. Wu, "Multi-objective community detection in complex networks," *Applied Soft Computing*, vol. 12, no. 2, pp. 850–859, 2012.

[31] M. Gong, L. Ma, Q. Zhang, and L. Jiao, "Community detection in networks by using multiobjective evolutionary algorithm with decomposition," *Physica A*, vol. 391, no. 15, pp. 4050–4060, 2012.

[32] M. Gong, B. Fu, L. Jiao, and H. Du, "Memetic algorithm for community detection in networks," *Phys Rev E*, vol. 84, no. 5, p. 056101, 2011.

[33] A. Rezvanian, B. Moradabadi, M. Ghavipour, M. M. D. Khomami, M.R. Meybodi, Social Community Detection. In Learning Automata Approach for Social Networks (pp. 151-168). Springer, 2019.

[34] R. Hosseini, A. Rezvanian, AntLP: ant-based label propagation algorithm for community detection in social networks. CAAI Transactions on Intelligence Technology, vol. 5, no. 1, pp.34-41, 2020.