Utilizing Cellular Learning Automata for Finding Communities in Weighted Networks

Mohammad Mehdi Daliri Khomami [a], Alireza Rezvanian [b], Ali Mohammad Saghiri [a], Mohammad Reza Meybodi [a]
[a] Soft Computing Laboratory, Department of Computer Engineering, Amirkabir University of Technology
[b] Department of Computer Engineering, University of Science and Culture, Tehran, Iran
m.daliri@aut.ac.ir, rezvanian@usc.ac.ir, a_m_saghiri@aut.ac.ir, mmeybodi@aut.ac.ir

Abstract—**The tremendous increase in Web usage led to the appearance of different network structures. One of the essential issues in the field of network science and engineering is to find and utilize network structures such as community structures by community detection. Although most of the current algorithms for detection of community use on the binary representation of the networks, some networks can encode more information instead of the topological structure, in which this information can be applied appropriately in detecting communities. Network information can be represented in the form of weights and identified as the weighted social network. This paper proposes a new algorithm based on irregular CLA (cellular learning automaton) for finding the community in weighted networks called CLA-WCD. The CLA-WCD can find near-optimal community structures with reasonable running-time by taking advantage of the parallel capability and learning ability of the cellular automata and learning automaton, respectively. The CLA-WCD is also evaluated on real and synthetic networks in comparison with popular community discovery methods. The simulation results demonstrated that the CLA-WCD outperforms other methods.**

Keywords—Weighted Community Detection; Social Network Analysis; Learning Automaton; Cellular Learning Automata.

I. Introduction

A lot of practical domains, such as Web, internet, online social networks [1], technological networks [2], and co-citation networks, can be studied by graph theory in order to investigate or explore the features of such domains. A social network (SN) is generally a social construction and interpreted as a graph $G = (V, E)$, such that V represents the set of social actors (individuals or organizations), and E is the edge-set connecting pairs of vertices and represents the relationships between them. In a real-world application, social networks are not binary and hold information on how a connection is strong or weak. As an example, in co-author networks, nodes denote authors, and edges are assigned between two nodes if they have corporate in the same paper. In this network, the number of cooperation is treated to show how 'strongly' both authors collectively accomplish their research [3]. As another example, in telecommunication networks, the weights of edges indicate call durations between individuals along a fixed period. In these social networks, edges are inherently weighted. From this point of view, an SN can be represented with a weighted graph. One of the common characteristics of these networks is community structure, which is referred to as the sets of nodes that are more strongly linked internally than with the other part of the network. It implies that the network has specific natural divisions within its structure.

The detection of community problems can be studied as the problem of partitioning in graph application, which is known as one of the NP-hard problems. Since graphs represent the networks, the detection of communities is equal to graph partitioning. Furthermore, the detection community structure is of the significant importance of getting insight into both the structures and the functions of a network.

Many scholars have proposed many algorithms to reveal communities in the networks, including modularity optimization [3-7], random walk [4,5], hierarchical [4-6], and spectral clustering [6,7], and user profile clustering [8,9]. These algorithms are all popular and well-known, but as mentioned before, most of the networks are weighted. Hence these algorithms are difficult to apply to weighted networks. Among all community detection algorithms, C-Finder [11] and RAK [12] are well-known and effective ones. C-Finder uses the concept of k-clique as a community that is composed of the union of all k-cliques. The later, RAK algorithm gives a label for each node uniquely, and utilized propagation of the label to reveal the communities.

Recently, some research is investigated community discovery in networks with edge weights, such as Strength [16], COPRA [15], and AntLP [3]. The idea behind the COPRA is using RAK [14]. Therefore, comparing to RAK, it is not easy to converge to a fixed status while the propagation of the labels and the found communities are non-deterministic. In ref. [16], the authors utilized the strength of nodes and the degree of belonging concept to find the structures of communities in overlapping. Although, by growing the overlapping structures in the network, the efficiency of the strength decreases significantly. In the AntLP [3], the concept of similarity among the links of networks is considered by ant colony optimization and modularity optimization for finding the communities. However, it seems that the results are not proper for large size networks.

To overcome the mentioned significant shortcoming, we proposed a new algorithm using irregular CLA (cellular learning automaton) for the detection of communities in weighted Networks called (CLA-WCD). In the CLA-WCD, the irregular CLA (ICLA) isomorphic to the input graph is formed, such that each cell corresponding to a vertex of the network. By interaction between learning automaton (LA) in each cell and network as its environment, each LA updates the status of each cell, and progressively the set of communities finds by the CLA-WCD. Moreover, during the finding of the communities by the CLA-WCD, the edge weights help to form better communities. Therefore, the central novelty of the CLA-WCD is to incorporate the edge weights for community detection based on ICLA.

In this paper, we first adders the problem of detection of community in weighted networks and then find the communities based on CLA. the detailed contributions of the CLA-WCD are as follows:

978-1-7281-1051-6/20/$31.00 ©2020 IEEE

- We present a new algorithm based on ICLA for the detection of community in weighted graphs or networks.
- The efficiency of the CLA-WCD is investigated on some popular synthetic and real networks.
- The CLA-WCD is compared with some famous algorithms for the detection of communities.

In the following sections, a brief description of ICLA is given in section 2. In section 3, the CLA-WCD is presented. In section 4, the efficiency of the CLA-WCD is studied in comparison with popular algorithms for the detection of communities based on artificial and real networks. Finally, concluding remarks are given in section 5.

II. CELLULAR LEARNING AUTOMATA

A hybrid model of cellular automata (CA) and learning automata (LA) is called CLA, in which each cell of CA is equipped with one or more LA in order to determine the state of the cells. [13, 14]. The CLA is known as an appropriate computational model for several distributed problems. In the CLA, the local environment is formed by the cell of LA and its direct neighbors together. Based on the interaction between LA and its environment in CLA, the probability of action selection of the neighboring LA changes during the evolution of the CLA. The main component of CLA can be defined as follows: A d-dimensional CLA is a structure $A = (Z^d, \Phi, A, N, F)$ such that

1. Z^ddenotes a lattice of d-tuples,
2. Φ denotes a limited collection of states,
3. A denotes the collection of LAs, exists in each cell of CA,
4. $N = \{x_1, \dots, x_m\}$ denotes a limited subset of Z^d called a vector of the neighborhood, such that $x_i \in Z^d$ of the CA,
5. $F: \Phi_m \rightarrow \beta$ denotes the rule in local in the CLA, such that β values as reinforcement signals serve by LA.

An ICLA, as demonstrated in Fig.1, is a CLA, such that the limitation of regular grid structure in conventional CLA is eliminated. This variation is done due to many applications that can be appropriately modeled with ICLA. An ICLA can be characterized as a network, in which each node indicates a cell and equips with an LA. The LA exists in a cell defines its action (status) accoering to its probability of action selection vector. Similarly to CLA, a rule governs the operation of the ICLA. Thus, the reinforcement signal comes from the environment responds to the reward and punishment of each LA in each cell of the ICLA. Numerous solutions using both the LA and CLA are presented in recent years for social network analysis [10], [17-20].

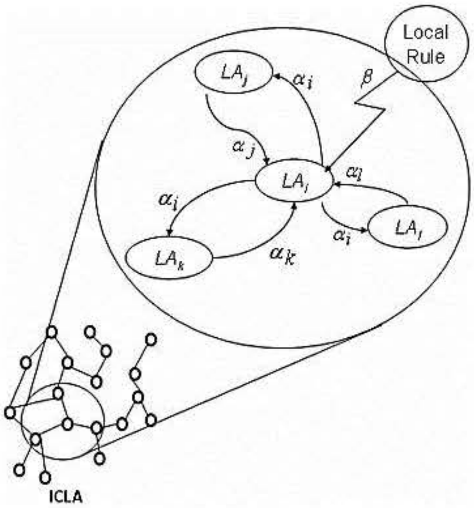

Fig. 1. A schematic structure of the ICLA [21]

III. CLA-WCD ALGORITHM

Let $G = (V, E)$ be a weighted graph, in which $V = \{v_1, v_2, \dots, v_n\}$ indicates the vertex-set, and $E \subseteq V \times V$ is the edge-set of the graph. In the weighted graph, w_{uv} is the edge weight between vertex $u \in V$ and vertex $v \in V$. The structure of k communities in a weighted graph can be represented as $C = \{c_1, \dots, c_k\}$ where $c_j \in C$ is the j^{th} community. For a node $u \in V$, k_u, is the degree of node and N_u is the set of neighboring node u, where $k_u = \sum_{v \in N_u} w_{uv}$.

The proposed algorithm, called CLA-WCD, utilizes a particular representation based on locus adjacency, which is introduced in [45] to determine the statuses of the cells of the ICLA. This representation has two advantages. First, there is no need for the size and density of communities, and this representation dedicates linear time complexity. In this locus representation, a candidate answer is represented by $S = (s_1, \dots, s_n)$ as candidate vector of solution which can be interpreted as s_j and node j is in an identical community. We note that the vector of the solution only represents the links instead of the communities. Since we are looking for representation for communities, therefore, a process is needed to transform the vector of solution into a vector of membership. To achieve this goal, a decoding mechanism is presented to show the structure of the community in the graph, which works over a depth-first search or breadth-first search as a linear time scale. Next to this process, the vector of solution is converted to the vector of membership that represents communities. Know we are ready to describe the whole process of weighted community detection.

For weighted community detection, the CLA-WCD, initially, an asynchronous ICLA is formed isomorphic to the input graph. In order to create the ICLA, each node in the graph is equipped with a cell of the CLA. Then an LA is allocated to each cell. The status of each cell specifies the current action taken by the LA exists in it. Each LA as LA_i exist in a vertex j is defined by a triple (α, β, P), where:

- $\alpha_j = \{\alpha_{j1}, \dots, \alpha_{jr}\} = N(j)$ indicates the collection of the action, and $N(i)$ represents the collection of the neighboring vertices of vertex j in the graph.

- $\beta_j \in \{0,1\}$ indicates the set of values for the response of the environment corresponds to reward and penalty.
- $P_j = (p_{j1}, p_{j2}, \dots, p_{jr})$ indicates the probability of action selection, where p_{ji} is the probability of action selection for action α_{ji} in L_j.

In each iteration, each LA chooses an action according to its corresponding probability of action selection. Based on the states of the cells, a solution vector S is constructed in which the current actions selected by all the LA. Therefore, in iteration t, the solution can be described by equation (1).

$$S(t) = \{\alpha_1(t), \alpha_2(t), \dots, \alpha_n(t)\} \quad (1)$$

where $\alpha_i(t)$ indicates the action selected by LA_i at iteration t. And then, the solution vector is converted to the membership vector. Moreover, the two following conditions are considered to check the quality of the obtained communities based on each automaton is rewarded or penalized.

1- Each node satisfies the condition that presented in [20], which indicates that the number of inside connections is no less than the number of links to that outside of the community. This condition applies a definition of the communities in which consider the topological characteristics of community structure.
2- The sum of the weights of the internal connections of the vertices, which belongs to the same community, is not less than the amount of the weights of the external vertices in others. Furthermore, mathematically can be described as (2):

$$W_i^{out}(C) < W_i^{in}(C), \qquad \forall i \in C, \quad (2)$$

where the $W_i^{in}(C)$ and $W_i^{out}(C)$ are the total inside weight and total outside weight of community i respectively, and defined by (3) and (4).

$$W_i^{in}(C) = \sum_{j \in C} W_{ij} \quad (3)$$

$$W_i^{out}(C) = \sum_{j \notin C} W_{ij} \quad (4)$$

We point out that if both conditions did not satisfy, then the corresponding automaton is penalized otherwise rewarded. The CLA-WCD proceeds until some successive iteration. The pseudo-code of the CLA-WCD is shown in Fig2.

IV. Analysis of Time Complexity

For the analysis of the time complexity of the CLA-WCD, assume that n be the number of vertices, m be the number of links, and T be the maximum iteration number. The degree in average is therefore $O(K) = O(m/n)$. The main step of the CLA-WCD lies in step 2 to step 4 as the rest of the steps can be done in the linear time scale as $O(n)$.

In step 2, the number of actions for each LA is equivalent to the number of connections of the vertex in the graph. Hence this step takes $O(K).O(n) = O(m)$. In step 3, the conversion solution vector to membership takes $O(n)$. In step 4, checking the defined constraint and total weight of communities for each node takes $O(k).O(n) = O(m)$. Therefore, the time complexity for T iterations is $O(T.M)$.

Algorithm 1. CLA-WCD

Input
  $W_{n \times n}$: the adjacency matrix of the input graph $G = (V, E, W)$, where W is edge weights of the graph
Parameters
  α: Reward parameter for the update of the action probability vector, where $0 < \alpha < 1$.
  r_i: The number of actions for learning automaton L_i, which is equal to the degree of node i in the network.
Initialization
  1) $p_{ij} = 1/r_i$, for $1 \le i \le n$ and $1 \le j \le r_i$.
Repeat
  2) Each L_i selects $\alpha_i(t)$ based on its $p_i(t)$, for $1 \le i \le n$.
  3) The solution vector $S(t)=(\alpha_1(t), \alpha_2(t), \dots, \alpha_n(t))$ is transferred into the membership vector $C(t)=(c_1(t), c_2(t), \dots, c_n(t))$ to represent the obtained structure of community by the process of decoding.
  4)
  Foreach L_i ($i \in \{1..n\}$) **Do**
    If $k_i(c_i(t)) \ge k_i(c'), \forall c' \ne c_i(t)$ & $W_i^{in}(C) > W_i^{out}(C), \forall i \in C$
      $\beta_i(t) \leftarrow 0$
    Else
      $\beta_i(t) \leftarrow 1$
    Endif
  Endfor
  5) Update the probability of action selection p_i of L_i.
  6) Until The found the structure of community remains unchanged
Output:
  The found communities, $C(t)$, and $S(t)$

Fig. 2. Pseudo-code of the CLA-WCD for detection of community in weighted social network

V. Experiments and Results

We evaluate the effects of the CLA-WCD in both real and synthetic LFR [22] networks. LFR benchmark parameters indicate the number of vertices, N indicates degree in average, k indicates the minimum vertex degree, max_c is the maximum size of communities, min_k specifies the minimum size of communities, and M indicates the parameter of mixing. Each node reflects a division of $1 - M$ links with other nodes in its community and a portion of M with other vertices of the input graph. In Table I, the information about the real graphs that are used for experiments. Besides, the CLA-WCD is compared with several algorithms for the detection of the community, such as Walk Trap [8], WCNM [23], and modularity optimization-based method [24].

TABLE I. THE REAL AND SYNTHETIC GRAPHS

Graphs	|V|	|E|
Karate	34	78
Les misérables	77	254
Neural Network	297	2148
LFR	1000	972

A. Weighted modularity

Due to nature of weighted graphs or networks, a structure of the community is not only associated with the density of edge among vertices however also the closeness of their connections, but Newman has also extended the modularity Q to modularity for the weighted graph as Q_w Moreover, defines as follows:

$$Q_w = \frac{1}{2W}\sum_{ij}\left(W_{ij} - \frac{w_i w_j}{2W}\right)\delta(c_i, c_j), \quad (5)$$

where N indicates the number of vertices, W_{ji} denotes the weight of edges between node j and vertex i, $W_i = \sum_j W_{ij}$ denotes the vertex strength of vertex i, and $2W = \sum_{i,j} W_{i,j}$ is the total strength of the vertices. c_i indicates the membership of vertex i to the i^{th} community c_i. Besides, the $\delta(c_i, c_j)$ the function returns one if the node j and node i are placed in the same community and 0, otherwise. Finally, Q_w, returns the total portion of weights of the edges when ending links occurred in the same community minus random case.

B. Normalized Mutal Information (NMI)

Another common evaluation metric for the structure of communities is normalized mutual information (NMI). The (NMI) usually computes the similarity between the correct community structures and the detected ones by the proposed algorithm in the network.

$$NMI(X, Y) = \frac{2I(X, Y)}{H(Y) + H(X)} \quad (6)$$

where $I(X, Y)$ indicate the mutual information and calculate the information by variables X and Y. $H(Y)$ is the entropy of the community of Y. NMI returns a value between [0, 1]. NMI measure is helpful when there is prior knowledge about the structure of communities.

C. Experiments

1) Experiment I

This experiment aims to investigate the efficiency of different learning models of the CLA, which are designed for this problem and select a proper model based on the obtained results for experiments. To do this experiment, we applied four different learning scheme including L_{R-I}, L_{R-p} and $L_{R-\varepsilon p}$.

We note that for simplicity Algorithm 1 for L_{R-I}, Algorithm 2 for L_{R-p} and Algorithm 3 for $L_{R-\varepsilon p}$ are applied. These algorithms are applied for real networks, and the obtained result is depicted in Fig. 3.

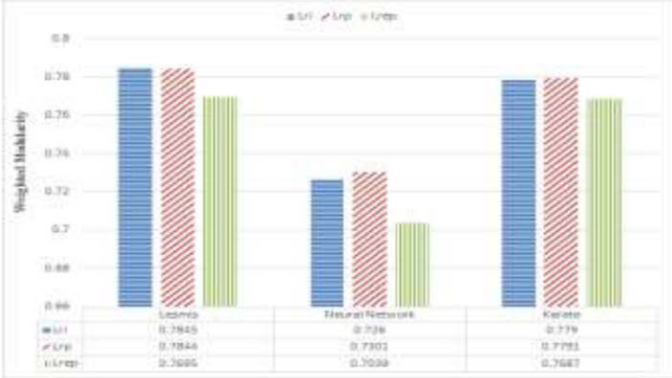

Fig. 3. The results of different learning algorithms applied in CLA-WCD in terms of weighted Modularity.

Based on the results, we may show that the second algorithm with a learning rate $a = 0.05$ and $b = 0.001$ are superior in most cases from others in terms of weighted modularity. Therefore, in the following experiments, algorithm two is applied as the default algorithm for comparison.

2) Experiment II

The experiment aims to compare the CLA-WCD with CLA-WCD, where the learning automata exist in each node is changed with a pure chance learning automaton (CLA-Pure). In pure chance learning automata, the action is selected with equal probability. Moreover, for simplicity, we conducted our experiment on synthetic LFR network. The result for both CLA-WCD and CLA-Pure is reported in Fig 4.

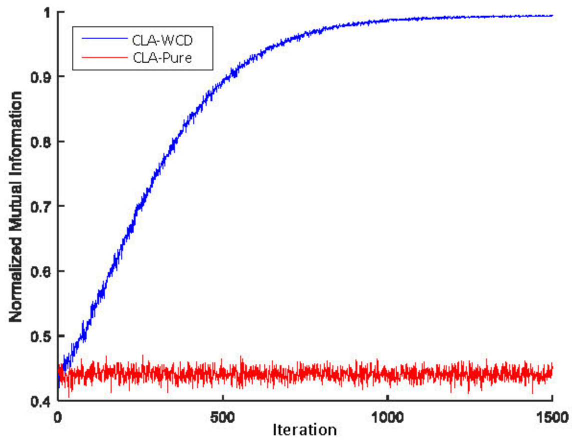

Fig. 4. Comparison of the CLA-WCD and CLA-Pure.

The impact of the LA in controlling the process of finding communities is illustrated in Fig.4. Using LA, the value of NMI for detected communities in the networks is higher where the learning is absent.

3) Experiment III

This experiment illustrates the obtained result of CLA-WCD compared with other algorithms such as Modularity optimization, WCNM, and Walk-trap. The result is reported 30 times independent execution by CLA-WCD in terms of NMI.

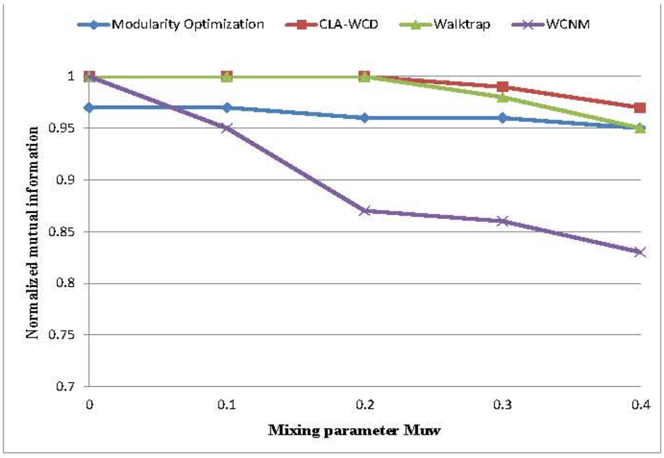

Fig. 5. The comparison of the algorithms for different values of mixing parameter

As can be seen in Fig 5, when the parameter of mixing is less than 0.2, the CLA-WCD and walk trap are the same in NMI when the mixing parameter precedes the CLA-WCD achieves significant results than other algorithms.

4) Experiment IV

The aim of this experiment is to carry out to study the efficiency of the CLA-WCD in comparison with different algorithms concerning weighted modularity. We note that the Best and average result is reported. As can be seen, the higher value is indicated in boldface in Table II.

TABLE II. WEIGHTED MODULARITY OF CLA-WCD AND WITH OTHER ALGORITHMS ON REAL SOCIAL NETWORKS.

Data Sets	MWEP	Walktrap	WCNM	WCLA-net		
	WQ	*WQ*	*WQ*	*WQ*		
				Best	*Avg*	*Std*
Karate	0.614	0.440	0.601	**0.775**	0.779	0.000
Lesmis	0.623	0.5402	0.788	**0.801**	0.798	0.003
Neural-Net	0.623	0.462	0.683	**0.750**	0.750	0.000
Net-science	0.824	0.369	0.953	**0.981**	0.960	0.002

As can be seen from the result in Table 2, we may conclude that the CLA-WCD due to considering two different conditions for finding weighted communities achieve a better result. These conditions guarantee that the algorithm detects communities not only by defining dense structures but also by the weight function is considered.

VI. Conclusion

To sum up, we proposed a novel algorithm based on ICLA for weighted community detection called CLA-WCD. Based on the CLA-WCD, the whole network is represented as irregular CLA, and the result is established by current actions selection by the LA in the graph. At each iteration, every LA selects one action by considering its probability of action selection and evaluated by considering the internal density of communities and weights. Then in the ICLA updates the probability of action selection according to the feedback from its environment. The efficiency of the CLA-WCD was examined by simulation against the popular methods for the detection of communities. The experimental simulations demonstrated the superiorities of the CLA-WCD concerning NMI and weighted modularity.

References

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

[2] J. Ruan and W. Zhang, "An Efficient Spectral Algorithm for Network Community Discovery and Its Applications to Biological and Social Networks," in 7th IEEE Int. Conf on Data Min (ICDM 2007), 2007, pp. 643–648.

[3] 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, 34-41, 2020.

[4] P. Basuchowdhuri J. Chen, "Detecting Communities Using Social Ties," in 2010 IEEE Int. Conf on Granul Comput, 2010, pp. 55–60.

[5] M. Girvan 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.

[6] M. E. Newman, "Fast algorithm for detecting community structure in networks," Phys. Rev. E, vol. 69, no. 6, p. 066133, 2004.

[7] B.-J. Cai, H.-Y. Wang, H.-R. Zheng, H. Wang, "Evaluation repeated random walks in community detection of social networks," in 2010 Int. Conf on Mach. Learn and Cybern, vol. 4, pp. 1849–1854, 2010.

[8] P. Pons, M. Latapy, "Computing Communities in Large Networks Using Random Walks," in Comp and Inform Sciences - ISCIS 2005, Berlin, Heidelberg, 2005.

[9] J. Chen, Y. Saad, "Dense Subgraph Extraction with Application to Community Detection," IEEE T Knowl Data En, vol. 24, no. 7, pp. 1216–1230, 2012.

[10] A. Rezvanian, B. Moradabadi, M. Ghavipour, M. M. Khomami, M. R. Meybodi. Learning Automata Approach for Social Networks. Cham, Switzerland: Springer; 2019.

[11] E. Jaho, M. Karaliopoulos, I. Stavrakakis, "ISCoDe: A framework for interest similarity-based community detection in social networks," in 2011 IEEE Conf on Comput Commun Work (INFOCOM WKSHPS), 2011, pp. 912–917.

[12] S. Alsaleh, R. Nayak, Y. Xu, "Finding and Matching Communities in Social Networks Using Data Mining," in 2011 Int. Conf on Adv in Soc Netw Anal Min, 2011, pp. 389-393.

[13] A. Rezvanian A. M. Saghiri, S. M. Vahidipour, M. Esnaashari, M. R. Meybodi, Cellular Learning Automata. In Recent Advances in Learning Automata 2018 (pp. 21-88). Springer, Cham.

[14] A. Rezvanian A. M. Saghiri, S. M. Vahidipour, M. Esnaashari, M. R. Meybodi. Recent advances in learning automata. Springer, 2018.

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

[16] R. Vafashoar, 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.

[17] A. Rezvanian, B. Moradabadi, M. Ghavipour, M. M. Khomami, M. R. Meybodi. Wavefront Cellular Learning Automata: A New Learning Paradigm. In Learning Automata Approach for Social Networks 2019 (pp. 51-74). Springer, Cham.

[18] A. M. Saghiri, M. M. Daliri Khomami, M. R. Meybodi, Intelligent Random Walk: An Approach Based on Learning Automata. Springer, Cham.

[19] M. Ghamgosar, M. M. D. Khomami, N. Bagherpour, M. R. Meybodi, "An extended distributed learning automata based algorithm for solving the community detection problem in social networks," in 2017 Iranian Conference on Electrical Engineering (ICEE), 2017, pp. 1520–1526.

[20] 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, vol. 24, pp. 413–426, 2018.

[21] M. Esnaashari, M. R. Meybodi, "Irregular cellular learning automata and its application to clustering in sensor network," in 15th Conference on Electrical Engineering (ICEE2007), Tehran, Iran, 2007.

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

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

[24] X. S. Zhang, R. S. Wang, Y. Wang, J. Wang, Y. Qiu, L. Wang, L. Chen "Modularity optimization in community detection of complex networks," Europhysics Letters (EPL), vol. 87, Aug. 2009.

[25] 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.