Solving Minimum Dominating Set in Multiplex Networks Using Learning Automata

Mohammad Mehdi Daliri Khomami
Department of Computer Engineering, Amirkabir University of Technology
Tehran, Iran
m.daliri@aut.ac.ir

Alireza Rezvanian
Department of Computer Engineering
University of Science and Culture
Tehran, Iran
rezvanian@usc.ac.ir

Ali Mohammad Saghiri
Information and Communications Technology Research Group, Niroo Research Institute
Tehran, Iran
amsaghiri@nri.ac.ir

Mohammad Reza Meybodi
Department of Computer Engineering
Amirkabir University of Technology
Tehran, Iran
mmeybodi@aut.ac.ir

***Abstract*—The dominating set (DS) problem has noticed the selecting a subset of vertices that every vertex in the graph is either is adjacent to one or more nodes of this subset. The DS with the minimum cardinality is called MDS (minimum dominating set). The MDS problem has several applications in different domains, such as network monitoring, routing, epidemic control and social network. The MDS is known as the NP-Hard problem. Nevertheless, the existing research has focused on the MDS problem to single networks. However, in many real structures, there exist a complex structure involving a set of components combined up by different connections and known as multiplex networks. In this paper, we introduce a learning automaton (LA) based algorithm for find the MDS problem in multiplex networks. In the proposed algorithm, each node of the multiplex network is considered an LA with two actions of a candidate or non-candidate corresponding to the dominating set and non-dominating set. By selecting candidate DS and evaluation mechanisms, the algorithm tries to find a dominating set with the smallest cardinality and as the algorithm proceeds, a candidate solution converges to the optimal solution of the MDS of multiplex networks. With the aid of learning and the behavior of learning automata for finding solution, this algorithm which is present in this paper reduces the number of dominating set, in multiplex networks iteratively. Experimental results demonstrate that in many well-known datasets, the proposed algorithm is efficient with respect to the evaluation measure.**

Keywords— Multiplexx Social Network. Dominating set. Learning Automata. Cellular Learning Automata

I. INTRODUCTION

In graph theory, a dominating set for a graph $G = (V, E)$ is subset D of V such that every vertex not in D is adjacent to at least one member of D. A dominating set with the minimum cardinality is called a minimum dominating set (MDS)[1]. The dominating set (DS) problem and its variations are an essential general class of optimization problems that many researchers widely notice[1]. In wireless networking, dominating sets are used to find efficient routes within ad-hoc mobile networks.

They have also been used in document summarization, and in designing secure systems for electrical grids. Recently, widespread applications of MDS have been found in many domains including ad hoc networks[2], wireless sensor networks[3], text summarization[4] and many other domains. Recently, many researchers have focused on to solve MDS problem and proposed a wide range of algorithms have been introduced, such as exact [5], [6] (exponential-time) algorithms, greedy algorithms[7]–[9]. and heuristic algorithms[10], [11] like ant colony optimization [12], genetic algorithms[13], simulated annealing [14]. However, these algorithms are studied dominating set problem for single layer networks, which may not proper for modelling the real structure of the networks. Due to multiple types of relationships in many real phenomena, multiplex networks may be better candidate for modelling real networks [15]. These networks are created by several single network, each of which contain the identical set of nodes yet different intra-layer edge and represent one type of connection between the nodes. In the real world, several networks are modeling in the form of a multiplex network such as biological networks, transportation networks, and social networks [16].

By taking into consideration the importance of multiplex social networks and the dominant minimum set problem, it is essential to tackle the MDS in multiplex networks due to its many applications in different fields. In this paper, we propose a learning automata-based algorithm for solving MDS problem in multiplex networks. The proposed algorithm consists of several stages which is first a network of learning automata is created and mapped to multiplex networks. Each leaning automata has two actions corresponds to be DS or not. In the second step, each learning automata select their action independently. The process of action selection is continued until to achieve a candidate solution by the action selection mechanism. After the second step is down. the cardinality of candidate DS is evaluated. If the cardinality of candidate DS is less than the DS which is obtained by algorithm, then the selected actions are rewarded else penalize. With the aid of the learning mechanism and learning automata, the process of finding the MDS is continued until to reveal near optimal solution.

The rest of the paper is organized as follows: section 2 gives the definition of MDS in single and multiplex networks and review the related works. In section 3, the learning automaton is introduced in brief. In section 4, our algorithm is described in detailed. The algorithm is simulated and analyzed with other classical algorithms in Section 4. At last, section 5 gives the concluding our paper.

II. Related works

A minimum dominating set MDS is a dominating vertex set of a graph containing the smallest possible number of vertices for the given graph. From simple to advanced algorithm for MDS, many algorithms reveal valuable information for finding MDS of social networks. An exact algorithm with exponential time complexity was the fastest algorithm on the MDS problem exploring all possible combinations. Based on the exact algorithm and using matching techniques the search space $Randerath$ and $Schiermeyer$ present an $O(1.8999^n)$ algorithm [17]. After that, *Grandoni* then reduced the time complexity to $O(1.8019^n)$[7]. A dived conquer technique, for MDS problem is presented with time complexity $O(1.4864^n)$ in [8]. The exact algorithm with exponential time complexity can be applied for an approximately small networks, which is not feasible in practical applications.

Another category of the algorithms that have been developed are greedy and heuristic to achieve near-optimal solutions. The naive greedy algorithm comprises successively picking nodes that could include the most crucial number of unvisited neighbors. Sanchis[10] proposed different greedy strategy, including GreedyRev, GreedyRan, GreedyVote, and GreedyVoteGr for finding DS in the networks. Due to complexity in selecting feasible nodes as the DS these algorithms are not appropriate for real applications. Zhao et al [11] demonstrated that networks without core, the MDS problems could be found precisely through the generalized leaf-removal strategy. Moreover, they also introduced a novel policy to construct dominating sets for the network with any degree distribution using the replica-symmetric mean-field theory. The simulated annealing techniques is used for finding a solution of dominating set with a specified size[14]. Also, a genetic algorithm that uses a 0-1 coding representation for the MDS is proposed by *Alkhalifah et al*[13]. Also, the ant colony optimization based algorithm is also proposed for MDS problem[18], [19].

In the real-world system, there exist widely multi-type relationships amongst the elements of the system. As an example, in social networks, people can communicate through many different mediums, such as face-to-face, phone calls, e-mail, and so on. The sophisticated modeling of many real worlds, modeling these networks as multiplex networks, maybe proper modeling for obtaining real-world systems' high-level complexity. The multiplex network contains the different layers in which each layer includes the equivalent number of nodes, and each node cannot be interconnected with other nodes in other layers.

The multiplex networks have a meaningful structure and dynamic characteristics from that in single networks[20]. In[21], [22]applied the extension of centrality measures their effect for a single layer to the multiplex. In [23], the walk, path, and length concepts are defined by changing the definition for a single layer to multiplex networks. In[24] studied the correlations between the roles of the nodes in different layers. Zhao et al, [25]demonstrated that the epidemic model could spread to a large scale in the multiplex network even if all the network layers are considerably below their particular epidemic thresholds. Moreover, in [24], the impact of correlations between different network layers that influence the spread of the epidemic outbreak is studied. In [26], Sole-*Ribalta et al*. have shown that the topology of multiplex networks cause the spread of influence for network flows in different layers of networks. In [27] have shown that the multiplex network's negatively correlated behavior is vulnerable to random failure, but it can be robust toward targeted attacks. In [16]explained that the communications among nodes in one layer could affect the cluster synchrotron in other layers of networks.

By considering the functionality and structural importance of multiplex networks and the importance of the dominant minimum set problem, in the following, we define multiplex networks and the minimum dominant set problem in multiplex networks.

Definition 1. The undirected multiplex network with$G^\alpha = \{G_1 = (V_1, E_1), G_2 = (V_2, E_2). \dots, G_k = (V_\alpha, E_\alpha \quad)\}$networks consists of α set of networks where $G_i = (V_i, E_i)$ is represent a social network, If a particular node exists in more than one OSN. then this node is known as an overlapping user of the multiplex G^α. Generally, in the multiplex, each network has the same number of nodes. Hence, if a node $v \in G_i$ does not belong to G_j this node is added to G_j as an isolated node. Then for all nodes, interlayer edges connect corresponds across all the multiplex networks. Consequently, we consider the set of all nodes of the multiplex social networks as $V = \bigcup_{i=1}^{\alpha} V_i$ where $|V| = N$.

Definition 2. Minimum Dominating Set of a given a multiplex network G^α is to find a minimum number of nodes $S \subseteq V$ of G^α if: $v_n^l \in S$ each node is in S or adjacent one vertex of S.

III. Learning Automata

Learning automaton (LA) [28] is a decision-making machine with learning capabilities. LA perform a finite set of actions α. Each action α_i is evaluated by a probability p_i and evaluation produce a feedback β. The objective of LA is learning to find the optimal solution. Figure 1 illustrates the relationship between the LA and the it's random environment.

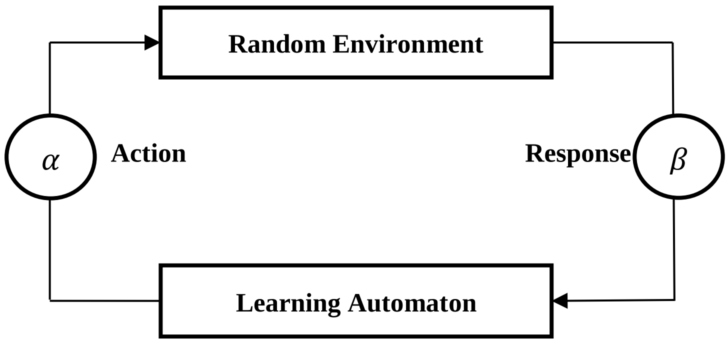

Fig. 1. The relationship between the LA and its probabilistic environment

LA is formulated as quadruple $\{\alpha.\beta.P.T\}$ where $\alpha = \{\alpha_1 . \dots . \alpha_r\}$ denotes the finite set of actions and $\beta = \{\beta_1 . \dots . \beta_m\}$ is the set of input values that can be taken by reinforcement signal and $P = \{p_1 . \dots . p_r\}$ is the probability vector of each action.

Equations (1) and (2) illustrate changes in probability vector according to performance evaluation of α in each step. Automata updates its action probability vector based on equation (1) for favorable responses as:

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

and equation (2) for unfavorable ones:

$$p_i(n+1) = (1-a)p_i(n)$$
$$p_j(n+1) = \left(\frac{b}{r-1}\right) + (1-b)p_j(n) \quad (2)$$

where $P(n)$ is the action probability vector at instant n. r is the number of actions that can be taken by the LA. The learning rates a and b denotes the reward and penalty parameters and determines the amount of increases and decreases of the action probabilities, respectively.[29]–[31].

IV. Proposed Algorithm

This section provides a learning automaton based algorithm for finding the minimum dominating set in the multiplex network (MDSM). The proposed algorithm tries to find a dominating set with the minimum cardinality. In the proposed algorithm before starting to finding the minimum dominating set, a set of learning automata which is isomorphic to the input multiplex network. This network can be defined by 2-tuple $\langle A^i . \alpha^i \rangle$ where $i \in \{1.2. \dots . \alpha\}$ and $A^i = \{A_1^i . \dots . A_n^i\}$ is the set of learning automata each of which assigned to a vertex of each nodes in layer i. $\alpha^i = \{\alpha_1^i . \alpha_2^i . \dots . \alpha_n^i\}$ is the action-set in which $\alpha_k^i = \{\alpha_{k.1}^i . \alpha_{k.2}^i . \dots . \alpha_{k.r}^i\}$ where $k \subset n$ is the set of actions where r_i is the number of actions that can be taken by learning automata A_k^i for each α_k^i.

The proposed algorithm consists of four main phases 1. Initialization, 2. Solution Finding, 3. Calculating objective function, and 4. Updating action probabilities. After the initialization is performed the algorithm iterates phases 2, 3, and 4 until the stopping conditions are met.

- **Initialization**

A network of learning automata $LA = \langle A^i . \alpha^i \rangle$ where $i \in \{1.2. \dots . \alpha\}$ isomorphic to the input multiplex network is mapped by assigning to each node of the multiplex network learning automata. The resulting network $A^i = \{A_1^i . \dots . A_n^i\}$ indicates the set of learning automata and $\alpha^i = \{\alpha_1^i . \alpha_2^i . \dots . \alpha_n^i\}$ shows the action set corresponds to automata A^i and $\alpha_k^i = \{\alpha_{k.1}^i . \alpha_{k.2}^i . \dots . \alpha_{k.r}^i\}$ where $k \subset n$ is set of action for learning automata A_k^i. We note that. A_k^i indicate learning automata k in layer i. Furthermore, each learning automata is assigned a probability vector and probability value for the selection of each action. A typical choice for the initial value of the probability vector is using uniform probability distribution. In the proposed algorithm, the action set of learning automata A_k^i in node v_k^i consist of two actions $\alpha_k^{i^0}$ and $\alpha_k^{i^1}$. Let φ_t^i be the selected candidate dominating set in iteration t. If $\alpha_k^{i^0}$ is selected then the corresponding node v_k^i be the member of φ_t^i. Otherwise, if $\alpha_k^{i^1}$ is selected no transform will happen in the association of the independent set.

- **Solution Finding**

During the execution of the algorithm, all learning automata concurrently take one of their actions according to the action probability vector. Based on the policy of the proposed algorithm. the action $\alpha_k^{i^0}$ for learning automata k in layer i shows that the current node v_k^i is the member of DS, while selecting $\alpha_k^{i^1}$ indicates that v_k^i is not be the member of DS. Therefore, all learning automata which select $\alpha_k^{i^0}$ and satisfying the dominating set condition be the member of φ_t^i for all : $i \in \{1.2. \dots . \alpha\}$. If not, the process of action selection by the learning automata is continued else; the next step is started. It is essential to note that, the process of action selection is done concurrently by all learning automata in all layers of the networks and make it possible to expedite the learning process.

- **Calculating objective function**

Based on the policy of the proposed algorithm, the action $\alpha_k^{i^0}$ for learning automata k in layer i shows that the current node v_k^i is the member of DS, while selecting $\alpha_k^{i^1}$ indicates that v_k^i is not be the member of DS. Therefore, all learning automata which satisfy the dominating set condition will have called candidate multiplex dominating set. It is essential to note that. the process of action selection is done concurrently by all learning automata in all layers of the networks and make it possible to expedite the learning process.

- Updating action probabilities

The objective function at iteration. $\varphi(t)$. be the average cardinality of last k iterations of candidate dominating set up to iteration t according to equation (3):

$$\varphi(t) = \frac{1}{k} \sum_{t=T-k}^{T} \sum_{i=1}^{\alpha} \varphi_t^i \quad (3)$$

Where T is the maximum number of iteration. φ_t^i is the candidate dominating set in layer i and iteration t. The objective function is stated that whether or not a dominating set found during an iteration can becomes a candidate or not. If the cardinality of dominating set during iteration has less than the objectives function $\varphi(t)$ then that dominating set is the candidate for multiplex network. otherwise it is not candidate.

- **Termination Conditions**

Up to iteration t. if the average cardinality of the dominating set is less than or equal to average dominating set till iteration t. then depending against the internal state. the actions chosen by all learning automata are rewarded. Each learning automaton updates its action probability vector by using L_{R-I} reinforcement scheme. Finding the minimum dominating set in the multiplex network and updating the action probabilities are continued until the number of iterations t exceed a predefined number T_{max} or $PC(t)$ defined by equation 4 reaches a predefined value P_{max}.

$$PC(t) = \prod_{i=1}^{\alpha} \varphi_t^i \quad (4)$$

$PC(t)$ is the product of maximum probability vectors of learning automata of the vertices of candidate dominating set at iteration t. Figure 2 presents the flowchart of Algorithm 1.

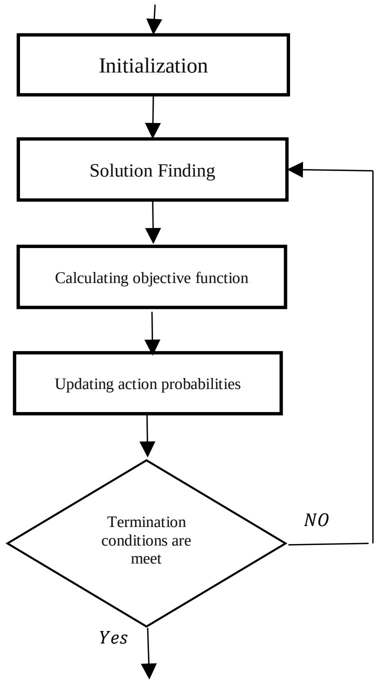

Fig. 2. Flowchart for the MDSM

V. SIMULATION RESULTS

We evaluated the efficiency of the proposed algorithm based on three well-known 2-layer multiplex datasets and is compared with several algorithms. Table 2 demonstrates the networks and their characteristics. These algorithms are implemented by MATLAB 2015b on a computer with 2.5 GHz Intel Core I 5 CPU and 4G memory running on Windows7 Operating system. Datasets are:

TABLE I. SPECIFICATION OF THE DATASETS FOR THE EXPERIMENTS

	#Node	#Edges
Dolphins	62	159
Scale Free 1	62	181
Email	1133	5451
Scale Free 2	1133	2266
Road EU	1174	1417
Scale Free 3	1174	2343

We note that. we have used the L_{R-I} learning algorithm with a 0.02 learning rate for the experiments.

- Experiment 1:

This experiment is designed to study the behavior of algorithms in terms of finding MDS and coverage loop. According to result in figure 3. as the algorithm progresses. the probability vector tends toward unity as the number of iterations increases. which can be concluded that the algorithm finds the dominant minimum set in multiplex networks.

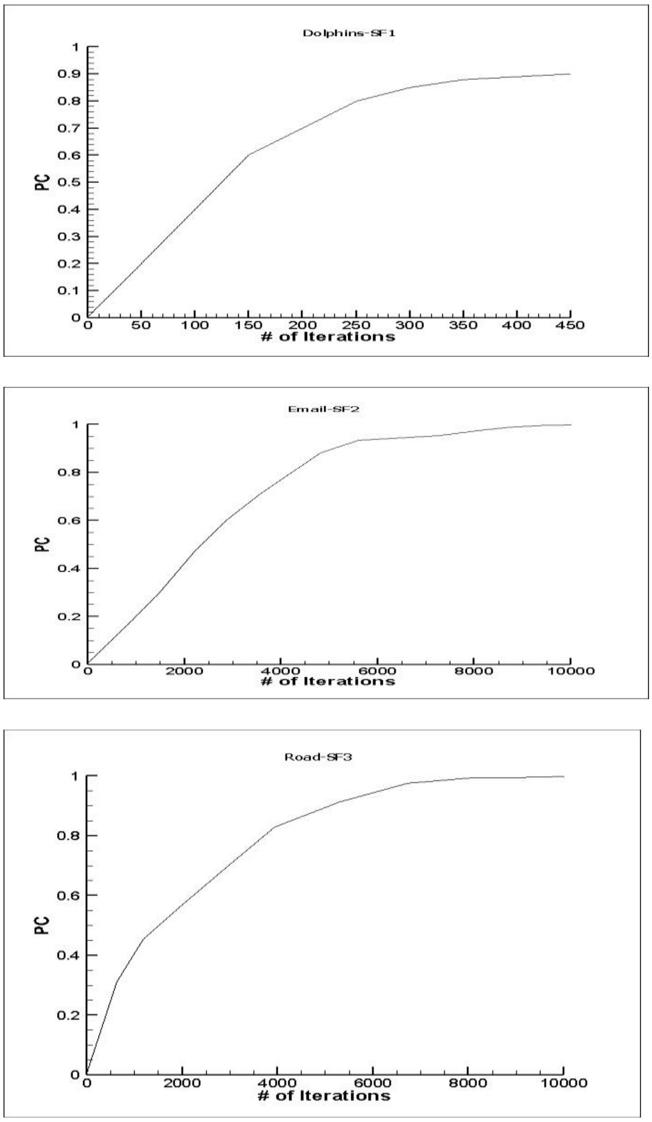

Fig. 3. The comparison of the behavior of algorithms in terms of finding MDS and coverage loop.

- Experiment 2:

This experiment is carried out to compare in terms of the number of dominant minimum nodes with Degree, greedy, simulated annealing (SA), and BDP algorithms. The result is shown in figure 3. In degree strategy, the algorithm considers nodes with the highest degree. and the degree of a node in a multiplex network indicates as the sum of its degrees across all the layers. Moreover, in Greedy strategy, the algorithms pick the node that has the largest number of edges connecting with undiscovered nodes in all layers. For SA and BDP please refer to [32].

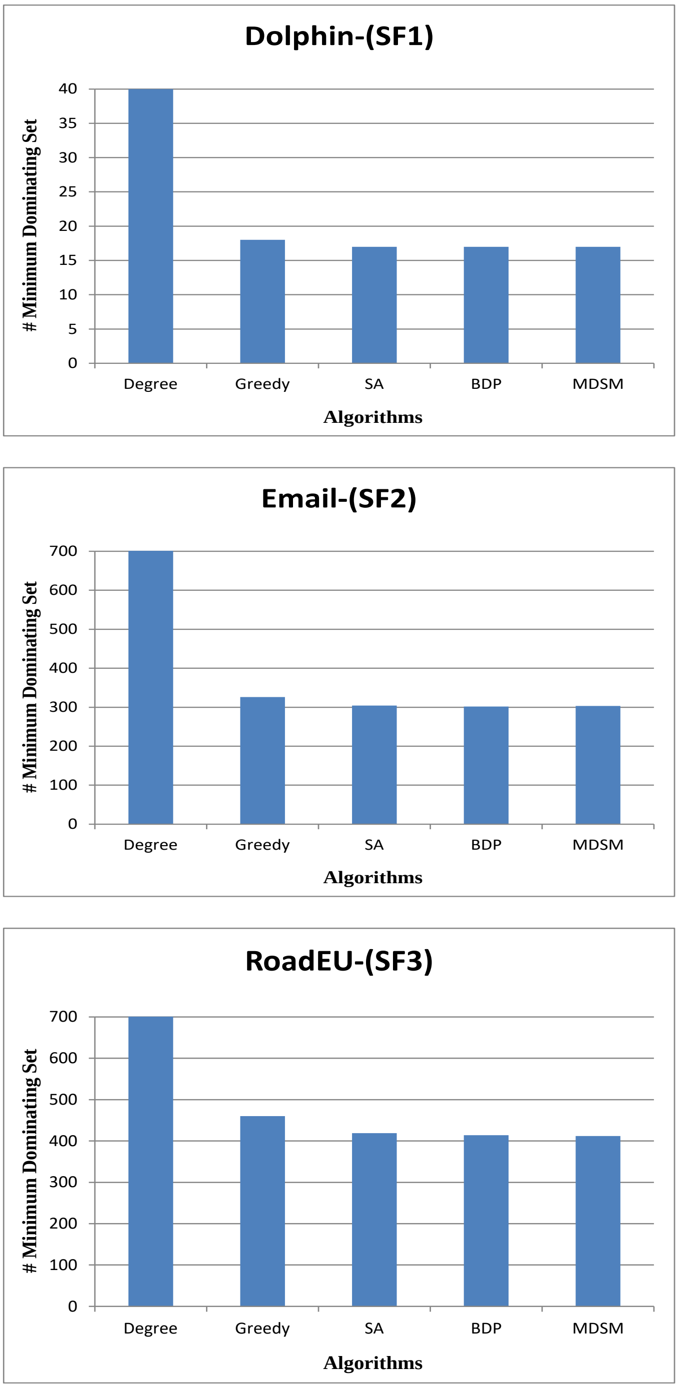

Fig. 4. Figure 4: The comparison of different algorithms in terms of number of dominating set in multiplex.

From the result, we may conclude that, in all networks the obtained result by MDSM is superior in terms of finding minimum dominating set in comparison with other algorithms. Since, a node with a large degree may have a small degree in a special layer and thus cannot let more nodes be observed in the layer. Hence, the degree may not be proper as good heuristic to obtained dominating set with minimum cardinality. Besides, due to the high complexity of the greedy algorithm. its precision is significantly more limited than that of the different algorithms like DLP and MDSM. With the aid of the learning mechanism, MDSM is looking for searching space for finding MDS.

VI. Conclusion

In this paper, we proposed learning automata based algorithm for finding dominating set with minimum cardinality in multiplex networks. The proposed algorithm consists of several stages in which the algorithm works including initialization, solution finding, calculate objective function and termination condition. In initialization phase, a network of learning automata is mapped to the input multiplex networks that each LA has two actions. In solution finding, a candidate solution is form based on definition of dominating set in multiplex networks. To evaluate the obtained solution, the cardinality of the solution is computed in calculate objective function and finally the algorithm based termination condition is stopped. To evaluate the performance of the algorithm, several experiment have been conducted and the obtained results shown the superior of the algorithm in terms of number of minimum dominating set with respect to other algorithms.

VII. References

[1] T. W. Haynes. S. Hedetniemi. and P. Slater. *Fundamentals of domination in graphs*. CRC press. 1998.

[2] J. Wu and H. Li. “A dominating-set-based routing scheme in ad hoc wireless networks.” *Telecommunication Systems*. vol. 18. no. 1–3. pp. 13–36. 2001.

[3] I. Stojmenovic. M. Seddigh. and J. Zunic. “Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks.” *IEEE Transactions on parallel and distributed systems*. vol. 13. no. 1. pp. 14–25. 2002.

[4] Y.-Z. Xu and H.-J. Zhou. “Generalized minimum dominating set and application in automatic text summarization.” in *Journal of Physics: Conference Series*. 2016. vol. 699. p. 012014.

[5] F. V. Fomin. D. Kratsch. and G. J. Woeginger. “Exact (exponential) algorithms for the dominating set problem.” in *International Workshop on Graph-Theoretic Concepts in Computer Science*. 2004. pp. 245–256.

[6] J. M. Van Rooij and H. L. Bodlaender. “Exact algorithms for dominating set.” *Discrete Applied Mathematics*. vol. 159. no. 17. pp. 2147–2164. 2011.

[7] F. Grandoni. “A note on the complexity of minimum dominating set.” *Journal of Discrete Algorithms*. vol. 4. no. 2. pp. 209–214. 2006.

[8] F. V. Fomin. F. Grandoni. and D. Kratsch. “Measure and conquer: domination–a case study.” in *International Colloquium on Automata. Languages. and Programming*. 2005. pp. 191–203.

[9] Y. Iwata. “A faster algorithm for dominating set analyzed by the potential method.” in *International Symposium on Parameterized and Exact Computation*. 2011. pp. 41–54.

[10] L. A. Sanchis. “Experimental analysis of heuristic algorithms for the dominating set problem.” *Algorithmica*. vol. 33. no. 1. pp. 3–18. 2002.

[11] J.-H. Zhao. Y. Habibulla. and H.-J. Zhou. “Statistical mechanics of the minimum dominating set problem.” *Journal of Statistical Physics*. vol. 159. no. 5. pp. 1154–1174. 2015.

[12] C. K. Ho. Y. P. Singh. and H. T. Ewe. “An enhanced ant colony optimization metaheuristic for the minimum dominating set problem.” *Applied Artificial Intelligence*. vol. 20. no. 10. pp. 881–903. 2006.

[13] Y. Alkhalifah and R. L. Wainwright. “A genetic algorithm applied to graph problems involving subsets of vertices.” in *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753)*. 2004. vol. 1. pp. 303–308.

[14] A.-R. Hedar and R. Ismail. “Simulated annealing with stochastic local search for minimum dominating set problem.” *International Journal of Machine Learning and Cybernetics*. vol. 3. no. 2. pp. 97–109. 2012.

[15] J. Gómez-Gardenes. I. Reinares. A. Arenas. and L. M. Floría. “Evolution of cooperation in multiplex networks.” *Scientific reports*. vol. 2. p. 620. 2012.

[16] S. Jalan and A. Singh. “Cluster synchronization in multiplex networks.” *EPL (Europhysics Letters)*. vol. 113. no. 3. p. 30002. 2016.

[17] B. Randerath and I. Schiermeyer. “Exact algorithms for minimum dominating set.” *Zentrum für Angewandte Informatik Köln. Lehrstuhl Speckenmeyer. Tech. Rep*. 2004.

[18] Q. S. ROMANIA. “Ant colony optimization applied to minimum weight dominating set problem.” in *Proceedings of the 12th WSEAS International Conference on Automatic Control. Modelling & Simulation. Catania. Italy*. 2010. pp. 29–31.

[19] A.-R. Hedar and R. Ismail. “Hybrid genetic algorithm for minimum dominating set problem.” in *International conference on computational science and its applications*. 2010. pp. 457–467.

[20] S. Boccaletti *et al.*. “The structure and dynamics of multilayer networks.” *Physics Reports*. vol. 544. no. 1. pp. 1–122. 2014.

[21] F. Morone and H. A. Makse. “Influence maximization in complex networks through optimal percolation.” *Nature*. vol. 524. no. 7563. p. 65. 2015.

[22] A. Solé-Ribalta. M. De Domenico. S. Gómez. and A. Arenas. “Centrality rankings in multiplex networks.” in *Proceedings of the 2014 ACM conference on Web science*. 2014. pp. 149–155.

[23] F. Battiston. V. Nicosia. and V. Latora. “Structural measures for multiplex networks.” *Physical Review E*. vol. 89. no. 3. p. 032804. 2014.

[24] D. Zhao. L. Li. H. Peng. Q. Luo. and Y. Yang. “Multiple routes transmitted epidemics on multiplex networks.” *Physics Letters A*. vol. 378. no. 10. pp. 770–776. 2014.

[25] D. Zhao. L. Wang. S. Li. Z. Wang. L. Wang. and B. Gao. “Immunization of epidemics in multiplex networks.” *PloS one*. vol. 9. no. 11. p. e112018. 2014.

[26] A. Solé-Ribalta. S. Gómez. and A. Arenas. “Congestion induced by the structure of multiplex networks.” *Physical review letters*. vol. 116. no. 10. p. 108701. 2016.

[27] B. Min. S. Do Yi. K.-M. Lee. and K.-I. Goh. “Network robustness of multiplex networks with interlayer degree correlations.” *Physical Review E*. vol. 89. no. 4. p. 042811. 2014.

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

[29] 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.

[30] 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.

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

[32] D. Zhao. G. Xiao. Z. Wang. L. Wang. and L. Xu. “Minimum Dominating Set of Multiplex Networks: Definition. Application. and Identification.” *IEEE Transactions on Systems. Man. and Cybernetics: Systems*. 2020.