



A Semi-supervised Multi-objective Evolutionary Algorithm for Multi-layer Network Community Detection

Ze Yin¹, Yue Deng¹, Fan Zhang¹, Zheng Luo¹, Peican Zhu²,
and Chao Gao^{1,2}(✉)

¹ College of Computer and Information Science, Southwest University,
Chongqing 400715, China

cgao@nwpu.edu.cn

² School of Artificial Intelligence, Optics, and Electronics (iOPEN),
Northwestern Polytechnical University, Xi'an 710072, China

Abstract. In the real world, many complex systems can be abstracted as multi-layer networks. Recently, community detection for multi-layer networks plays a vital role in multi-relationship complex system analysis, thus gradually gaining popularity especially in the optimization algorithms. The multi-objective optimization (MOOP) methods attract attention owing to the flexibility in solving community detection problems. Nevertheless, most of the MOOP methods pay little attention to the prior information, which cannot ensure the high-level accuracy and robustness against networks with complicated community structures. To address the problem, this paper proposes a semi-supervised multi-objective evolutionary algorithm for multi-layer community detection (SS-MOML). The SS-MOML mainly consists of two steps: First, it extracts the prior information from the network. Second, based on the prior information, the prior layer is constructed by creating virtual connections and the high-quality initial population is generated. And then the optimization process begins, in which the genetic operation based on the prior information is committed to guiding the evolutionary direction of chromosomes. Some extensive experiments are implemented and the results prove that the proposed SS-MOML stands out in accuracy and robustness than 7 state-of-the-art multi-layer community detection algorithms.

Keywords: Community detection · Multi-layer network ·
Semi-supervised multi-objective optimization · Prior information

1 Introduction

Networks have become effective tools for processing complex systems in the real world such as social networks, biological networks and technological networks [10]. In a network, nodes represent the entities, and edges stand for the connections between entities. The traditional single-layer network can only represent a

single relationship. However, the relationship of entities is diversified. Therefore, multi-layer networks, which can address multi-relationship more effectively, have gradually been a research hot spot [11].

Communities are one of the most essential attributes of networks, which can reveal the potential structure characteristics of networks [10]. Generally speaking, a community (module or cluster) is a series of nodes with closer connections or more similar attributes [6]. Up to now, single-layer network based community detection algorithms have made great achievements, for example, Gao et al. propose a physarum-based framework [5] and Yang et al. propose a semi-supervised framework [17]. However, because of the particular structure of multi-layer networks, traditional single-layer based methods cannot address multi-layer networks effectively. In the past decades, some multi-layer community detection algorithms have been proposed, among which multi-objective optimization algorithm (MOOP) is one of the most effective methods. Due to the local and global search capabilities of the evolutionary algorithm, the multi-objective evolutionary algorithm (MOEA), one of the most classical MOOP methods, has attracted great attention [9]. MOEAs achieve good performance because they can balance the information of each layer by optimizing different objective functions. However, this kind of algorithms pays much attention to the topological information rather than taking prior information into full consideration.

Yang et al. suggest that only considering the topological information is not adequate for algorithms [17]. The prior information can further improve the accuracy and robustness of the algorithm by dividing the preknown similar nodes into the same community despite the weak relationship of the topological information. For example, the prior information can be added to the non-negative matrix factorization-based (NMF) or spectral clustering-based (SC) algorithm as a graph regularization [17]. However, there are few researches for incorporating the prior information into the MOEA. This is because it is difficult to extract the consensus prior information due to the complex structure of multi-layer networks. Besides, unlike the NMF and SC methods, MOEAs find the optimal community partition by imitating the population evolution, which brings difficulties in applying the guidance of prior information.

To address the problem mentioned above, this paper proposes a novel **Semi-Supervised Multi-Objective** evolutionary algorithm for **Multi-Layer** network community detection (named SS-MOML). The main idea of SS-MOML is to incorporate the prior information into the MOEA from beginning to end. More specifically, it aims to construct the prior layer, initialization and prior information based genetic operation, which incorporates the prior information into networks and optimization process, respectively. The prior layer, high-quality initialization and modified genetic operation can guide the evolution of the population by dividing similar nodes into the same community as far as possible. The contributions are listed as follows.

- Instead of only paying attention to the topological information like the traditional MOEAs, the SS-MOML neatly combines the idea of semi-supervised

methods and MOEA methods on different aspects, so as to further improve the accuracy and robustness.

- In order to extract the prior information for multi-layer networks with various structures, this paper proposes DeepWalk-based method and *Sørensen – Dice’s* similarity-based method, where a novel density-based aggregation strategy is added into the DeepWalk to adapt to various network structures.
- In order to get the utmost of the prior information and MOEA, this paper proposes a SS-MOML algorithm, which utilizes the prior information to generate the prior layer, initial population and guide the population evolution.

The paper is organized as follows. Sect. 2 introduces the problem definition of multi-layer network community detection. The proposed SS-MOML algorithm is presented in detail in Sect. 3. The extensive experiments are elucidated in Sect. 4. Finally, Sect. 5 concludes the paper.

2 Problem Definition

A multi-layer network can be represented as a graph $G = (V, E_l (l = 1, \dots, L))$, which consists of L layers. Each layer has the same node set $V = \{v_1, v_2, \dots, v_N\}$ which satisfies $|V| = N$ and N represents the number of nodes. E_l stands for the edge set of the l_{th} layer which represents a kind of relationship. Existing approaches formulate the multi-layer network as a set of matrices, $A^{(l)} \in \mathbb{R}_+^{N \times N} (l = 1, \dots, L)$, which is used to encode the information of each layer.

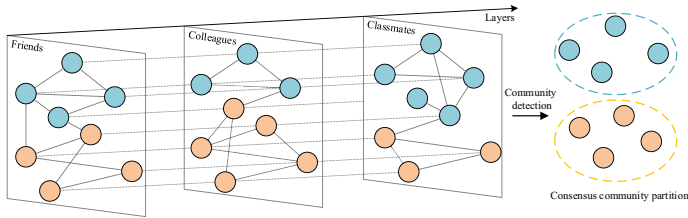


Fig. 1. An example of multi-layer networks. The network consists of 3 layers, and different layers represent different relationships (i.e., friends, colleagues, and classmates). Nodes pertaining to different communities are colored differently.

A multi-layer network has multiple layers, where each layer represents one relationship, which is shown in Fig. 1. The complex structure of multi-layer networks leads to the diversity of the community structure in each layer. The ultimate goal of the multi-layer network community detection algorithm is to find a consensus community partition that best adapts to each layer of the multi-layer network.

3 Proposed Method

The proposed SS-MOML algorithm is introduced in this section. The flowchart of the proposed algorithm is shown in Fig. 2. The algorithm mainly consists of two components, namely, the prior information extraction and the optimization process. In the former part, two prior information extraction methods are proposed, that is, DeepWalk-based method (shown in (a)) and *Sørensen – Dice’s*-similarity-based method (shown in (b)). The optimization process is based on the NSGA-II, and the prior information is incorporated into the optimization section in each step. The main components of the optimization process are shown in (c), and each of them is introduced as follows.

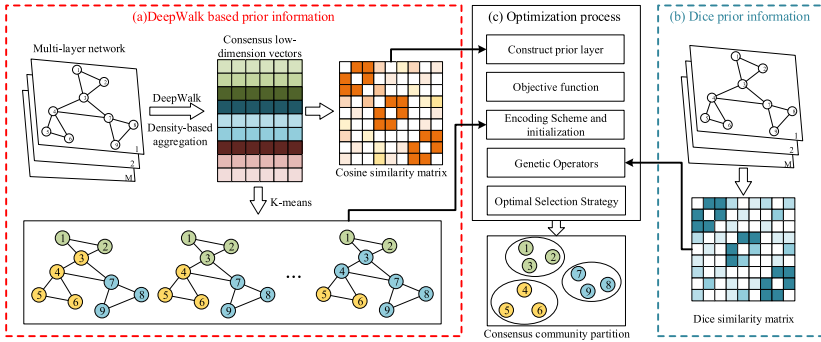


Fig. 2. The flowchart of the SS-MOML algorithm. (a) and (b) represent the DeepWalk-based and the *Sørensen – Dice’s*-based prior information extraction process, respectively. (c) is the optimization process. As shown in the figure, the DeepWalk-based information is utilized to construct the prior layer and the *Sørensen – Dice’s*-based information to guide the genetic operation.

3.1 Prior Information Extraction

As mentioned above, the prior information is used to guide the algorithm to divide the similar nodes into the same community. Therefore, the prior information plays a key role in algorithm since high accurate prior information can provide a correct guidance. For example, Ma et al. believe that the subgraph composed of closely connected nodes should be the prior information and they find such prior information through a greedy search method [10]. In this paper, we extract the prior information by applying DeepWalk (DW for short) [14] and *Sørensen – Dice’s* (Dice for short) similarity [16] because the DW can extract the relatively high-order information and Dice can eliminate the effect of node degree. Firstly, the DeepWalk algorithm is applied in each network layer to acquire the low-dimensional representation vectors of each one. Due to the particularity of multi-layer network structure (i.e., each network layer represents a certain kind of relationship), the low-dimensional vectors need to be merged

into a consensus vector as the consensus prior information of the network. The most intuitive way is to compute the average of each vector. However, the quantity of information and the importance of each layer are different, so calculating the average of vectors may lead to low accuracy. To overcome the problem, we propose the density-based aggregation strategy, which quantizes the amount of information in each layer using density, which functions as the weight to merge vectors from different layers [8]. Since the increase of density means the decrease of information quantity on the contrary, it is inconvenient to calculate the weight. To address this problem, the equation is converted to Eq. (1).

$$density = \frac{M_c}{N(N-1)/2} \quad (1)$$

where M_c represents the number of connections within communities, and N is the number of nodes. The density means the proportion of connections and nodes within a community. We regard a network layer as a community, so that the density can be used to quantize the information quantity of a network layer.

The second method is to extract the prior information by applying Dice similarity. The equation is shown in Eq. (2).

$$\mathbf{S}(v_i, v_j) = \frac{2ComNeighbours(v_i, v_j)}{Length(v_i) + Length(v_j)} \quad (2)$$

where v_i and v_j represent the nodes, $ComNeighbours(v_i, v_j)$ means the number of common neighbours of nodes v_i and v_j . $Length(v_i)$ is the degree of the node v_i . To preserve the structure of the multi-layer network, the neighbours of all layers are considered when calculating the number of neighbours of nodes, which ensures the connection information of all layers are considered.

After that, the prior information contains the similarity between every two nodes. To construct the prior layer, we retain a part of node pairs with the highest DW-based cosine similarity as virtual edges according to a predefined threshold θ . Finally, the collection of virtual edges constitutes the prior layer. The prior layer contains accurate information of community structure because the connected nodes are more likely to belong to the same community (they are more similar). In addition, the Dice-based prior information is incorporated into the genetic operation, which is described in Sect. 3.4.

3.2 Objective Functions

The MOEA can be represented as $F = \{F_1, F_2, \dots, F_t\}$, where F_i is the i_{th} objective function, and t means the number of functions. The MOEA optimizes each objective function to find the optimal solution. In this paper, we choose two widely-used objective functions, Modularity (Q) [12] and Normalized Cut (Nc) [3], which are shown in Eq. (3) and Eq. (4).

$$Q = \frac{1}{2M} \sum_{i,j}^N \left(A_{ij} - \frac{d_i \times d_j}{2M} \right) \delta(C^i, C^j) \quad (3)$$

where M is the number of edges, A represents the corresponding adjacent matrix and d_i means the degree of node i . C stands for the community and C^i represents the node i that belongs to C . When $C^i = C^j$, $\delta(C^i, C^j) = 1$, and 0 otherwise. In this paper, the first objective function is the average of Q for each layer.

$$Nc\left(\{C_k\}_{k=1}^K\right) = \frac{1}{K} \sum_{k=1}^K Nc_k \quad (4)$$

where $Nc_k = \frac{W_k^o}{2 \cdot W_k^i + W_k^o} + \frac{W_k^o}{2 \cdot (W_k^a - W_k^i) + W_k^o}$, and C_k is the k_{th} community. W_k^i , W_k^o and W_k^a are the sum of edge weights within communities, between communities, and total edges of community k , respectively. To adapt to the structure of multi-layer networks, the W_k^i , W_k^o and W_k^a are calculated for all layers.

3.3 Encoding Scheme and Initialization

The encoding scheme will affect the computational cost of the algorithm. Label-based and locus-based methods are two widely-used encoding approaches. However, the label-based method may cause redundancy, which leads to space consuming. For example, the encoding sequence $\{1\ 1\ 1\ 2\ 2\ 2\ 2\}$ and $\{2\ 2\ 2\ 1\ 1\ 1\ 1\}$ represent the same community partition. Therefore, the locus-based encoding scheme is selected in the proposed algorithm.

To generate the high-quality initial population, we apply the k-means algorithm to the consensus low-dimensional vector (calculated by DW-based method). Then the clustering result is transformed to the locus-based encoding, which is the initial population. The flowchart of initialization is also shown in Fig. 2.

3.4 Genetic Operators

Genetic operation is one of the most critical components, which is helpful to break the local optimality and find the optimal solution. In this paper, the crossover and mutation operations are used to increase the population diversity.

The crossover operation can generate various offspring chromosomes. The uniform crossover is selected in this paper because it is more random. Firstly, a binary mask, having the same length as the chromosome, is generated randomly. The offspring is generated by choosing the gene of parents according to the value of the binary mask. More specifically, at a gene, when the value of the mask is 0, the offspring chooses the first parent, otherwise another parent.

To make full use of the guiding ability of Dice-based prior information (calculated in Sect. 3.1), we propose the Dice-based mutation strategy. Simultaneously, to ensure the utilization of the local information (i.e., neighbours of nodes), we design a mutation strategy consisting of two parts as shown in Fig. 3.

There are two mutation strategies, and 50% of parent chromosome conducts the neighbour-based mutation strategy and others perform the Dice-based mutation strategy to take full advantage of the local information and prior information. For neighbour-based strategy, a gene of a parent chromosome is selected

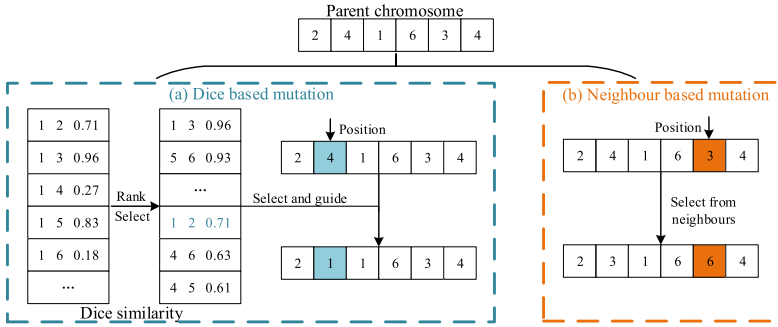


Fig. 3. The diagram of mutation operation. The mutation consists of two components, which are neighbour-based (red dotted line) and Dice-based strategy (blue dotted line). (Color figure online)

randomly according to a predefined probability. Then, the value of the selected gene mutates to the value of one of its neighbours. As for the Dice-based strategy, some genes are selected randomly with a predefined probability and the value of genes is selected from the Dice similarity table randomly. Similar to DW similarity mentioned in Sect. 3.1, the Dice similarity table is constructed by a part of node pairs with the highest Dice similarity according to a predefined reservation threshold ϵ . Figure 3 illustrates the processes of genetic operation.

3.5 Optimal Selection Strategy

As mentioned above, MOEAs can find a collection of solutions. Finding an appropriate way to extract the optimal solution is important for the algorithm. The knee point strategy is a useful method [1]. The idea of the knee point refers to a little improvement of one objective function leading to a significant reduction of other objective functions. For a two-fitness-function MOEA, an angle-based method is applied to find the knee point of the solution set [1].

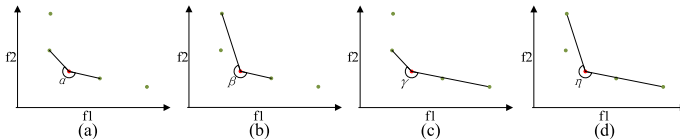


Fig. 4. An example of knee point strategy. The red point represents the target node and green points are four neighbours of the target node. α , β , γ and η are four angles. (Color figure online)

The angle-based method estimates the trades-offs of two objective functions by measuring the slopes of two lines, where the line is constructed by passing

through the node and its neighbours. In the angle-based measure, first, we select four neighbors, which are the closest or the second closest on either side of the target solution. Two neighbors on different sides are randomly chosen as a pair to calculate the angle with the target solution. The four angles are shown in Fig. 4. The point with at least one maximal angle can be designated as the optimal solution.

Although the knee point strategy can find the optimal solution, the total iteration times keep unchanged, which will lead to the fluctuation of the Pareto set. Therefore, a modularity-based selection strategy is used as a supplement. The strategy selects the solutions by finding the maximum average of modularity.

4 Experiment

4.1 Datasets and Metrics

Various synthetic and real-world datasets are used to validate the performance of the proposed algorithm. The basic information of datasets is shown in Table 1. The synthetic datasets are generated by the multi-layer LFR benchmark (mLFR) [2]. The mLFR benchmark controls the structure of the network by changing the mixing parameter (μ) and the degree change chance (Dc). Both the μ and Dc range in $(0, 1)$. With the increase of μ and Dc , the community structure of the multi-layer network becomes more complicated. The real-world networks with different types and scales (i.e., SND [7], MPD [7], WTN [7], CoRA [7], Citeseer [7]) are used to ensure the comprehensiveness of the experiment.

Table 1. The summary of datasets. The first five are real-world datasets and the last two are generated by the mFLR benchmark.

Network	Layers	Nodes	Clusters
SND	3	71	3
MPD	3	87	6
WTN	14	183	10
CoRA	2	1662	3
CiteSeer	2	3312	3
Syn1	3	5000	9
Syn2	3	10000	16

Besides the datasets, metrics are also significant for experiments. In this paper, two widely-used metrics are used to evaluate the accuracy of the detected communities, namely, Normalized Mutual Information (NMI) [7] and Adjusted Rand Index (ARI) [7]. Both two metrics evaluate the similarity between partition C and Ω , where Ω represents the ground truth and C stands for the partition calculated by the proposed algorithm. The range of NMI and ARI is $[0,1]$. The larger the both metrics, the more accurate the community partition.

Table 2. The experimental results on various datasets. The $\mathcal{A}_1 - \mathcal{A}_8$ represent SS-MOML (our proposed algorithm), MOEA-MultiNet, SC-ML, MIMOSA, S2-jNMF, COMCLUS, CSNMF and GMC algorithm, respectively. The ‘-’ represents that the algorithm cannot run successfully on this network.

	Algorithms	\mathcal{A}_1	\mathcal{A}_2	\mathcal{A}_3	\mathcal{A}_4	\mathcal{A}_5	\mathcal{A}_6	\mathcal{A}_7	\mathcal{A}_8
SND	<i>NMI</i>	0.721	0.437	0.681	0.132	0.582	0.555	0.681	0.597
	<i>ARI</i>	0.747	0.201	0.493	0.088	0.452	0.480	0.493	0.428
MPD	<i>NMI</i>	0.591	0.494	0.495	0.096	0.516	0.421	0.504	0.451
	<i>ARI</i>	0.468	0.385	0.379	0.01	0.396	0.365	0.394	0.248
WTN	<i>NMI</i>	0.316	0.140	0.231	0.065	0.157	0.183	0.284	0.203
	<i>ARI</i>	0.197	0.052	0.129	0.007	0.069	0.105	0.197	0.015
CoRA	<i>NMI</i>	0.886	0.317	0.480	0.011	0.796	0.471	0.514	0.519
	<i>ARI</i>	0.924	0.243	0.485	0.001	0.813	0.447	0.491	0.426
CiteSeer	<i>NMI</i>	0.345	-	0.191	-	0.149	0.182	0.237	0.042
	<i>ARI</i>	0.337	-	0.169	-	0.147	0.119	0.207	0.012
Syn1	<i>NMI</i>	0.965	-	0.950	0.855	0.772	0.950	1	0.999
	<i>ARI</i>	0.972	-	0.902	0.896	0.602	0.902	1	0.999
Syn2	<i>NMI</i>	0.953	-	1	0.098	0.452	0.894	0.946	0.963
	<i>ARI</i>	0.895	-	1	0.01	0.074	0.814	0.874	0.954

4.2 Baselines and Parameter Settings

In SS-MOML, the parameter settings of DeepWalk are the same as the source paper. The number of iterations and populations are both 500. Besides, θ and ϵ , which are used to control the proportion of retaining prior information, are set as 0.3 and 0.1, respectively. The analysis of parameters is shown in Sect. 4.4.

To verify the performance of our algorithm, various kinds of algorithms are selected as the comparison algorithms, namely, MOEA based method (MOEA-MultiNet [9]), spectral clustering based methods (SC-ML [4] and MIMOSA [3]), matrix factorization based approaches (S2-jNMF [10], COMCLUS [13] and CSNMF [7]) and multi-view clustering based method (GMC [15]). The parameter settings of comparison algorithms are consistent with the source paper.

4.3 Experimental Results

The results of comparison experiments are shown in Table 2. The experiment consists of seven comparison algorithms mentioned above running on five real-world datasets and two synthetic datasets (Syn1 and Syn2).

The results show that the proposed SS-MOML performs better than other state-of-the-art methods on most of the datasets. Since the prior information guides the algorithm to divide similar nodes into the same community, the proposed algorithm can keep a high-level performance on Citeseer, WTN and MPD, which have a complicated community structure. Moreover, we find that the SS-MOML has a higher accuracy than another classical MOEA-based method

(MOEA-MultiNet) on most networks due to the utilization of prior information. On Syn1 and Syn2 (large scale synthetic network), the accuracy of SS-MOML is slightly lower than some comparison algorithms because the structure of synthetic networks is simpler compared with real-world networks and the objective function values are not linearly related to the accuracy. In general, the performance of SS-MOML surpasses other algorithms on various kinds of multi-layer networks.

4.4 Parameter Analysis

In this section, some experiments of threshold θ and ϵ are implemented to verify the effects of two parameters on our algorithm. The results are shown in Fig. 5.

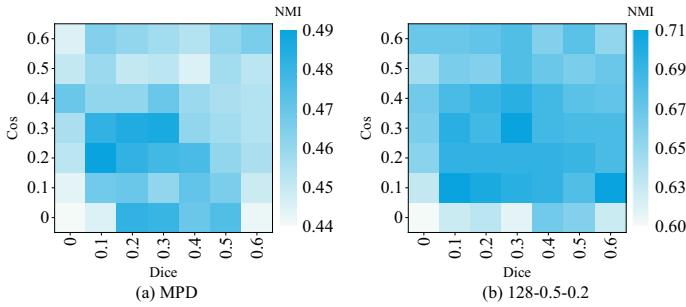


Fig. 5. The experimental results of parameter analysis. The x-axis and y-axis represent the threshold of Dice (ϵ) and cosine similarity (θ), which controls the reservation proportion of prior information. To ensure the comprehensiveness of experiments, a real-world network (MPD) and synthetic network (128-0.5-0.2) are chosen as experimental datasets. To avoid the fluctuation of knee point selection strategy, the experiments take the second optimal selection strategy (maximize the average modularity).

The experimental results indicate that the *NMI* value drops to the minimum on both two datasets when $\theta = 0$ and $\epsilon = 0$, which means there is no prior information incorporated into the algorithm. With θ and ϵ increase, the *NMI* value rises because the prior information guides the algorithm to find better solutions. As shown in (a), when the values of θ and ϵ surpass 0.5, the reserved quantity of prior information grows up, leading to the decrease of the accuracy for prior information (the similarity between nodes declines). The experiments prove the positive impact of prior information on the algorithm.

4.5 Robustness Analysis

As mentioned above, the prior information can improve the robustness of the algorithm. This section validates the robustness of the proposed SS-MOML algorithm. The experiment, as shown in Fig. 6, consists of two sub-experiments, which change the network structure and number of layers respectively.

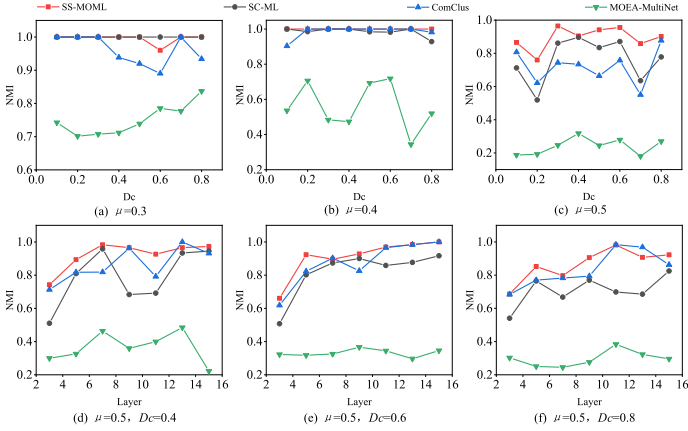


Fig. 6. The analysis of robustness. (a)–(c) are designed to validate the stability of algorithms for various network structures. (d)–(f) represent the experimental results running on networks with different layers.

The results show that the proposed SS-MOML performs better than other state-of-the-art methods in terms of accuracy and stability on various synthetic networks with different structure and number of layers. Compared with MOEA-MultiNet, the proposed algorithm owns a remarkable advantage. In addition, we can find that the NMI has less volatility than the other state-of-the-art methods with high robustness. It is because that the prior information participates in the initialization and each step of the MOEA, which can ensure the nodes with high similarity are distributed into the same community.

5 Conclusion

This paper proposes a novel semi-supervised multi-objective evolutionary algorithm for multi-layer network community detection, called SS-MOML. SS-MOML first extracts the prior information using DW-based strategy and Dice-based strategy. Then, it constructs the prior layer based on the DW-based strategy and the optimization process begins. In each iteration step, the Dice-based mutation strategy will guide the population evolution based on the Dice-based prior information. Moreover, the prior information can evaluate the similarity between nodes, and guide them into the same community despite the weak relationship on topological information. A series of extensive experiments prove our SS-MOML performs superiorly to other state-of-the-art algorithms on most datasets.

Acknowledgement. This work was supported by the National Key R&D Program of China (No. 2020AAA0107700), National Natural Science Foundation of China (Nos. 61976181, 11931015), Key Technology Research and Development Program of Science and Technology Scientific and Technological Innovation Team of Shaanxi Province

(No. 2020TD-013) and the Science and Technology Support Program of Guizhou (No. QKHZC2021YB531) and National Municipal Training Program of Innovation and Entrepreneurship for Undergraduates (No. S202110635042).

References

1. Branke, J., Deb, K., Dierolf, H., Osswald, M.: Finding knees in multi-objective optimization. In: Yao, X., et al. (eds.) PPSN 2004. LNCS, vol. 3242, pp. 722–731. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30217-9_73
2. Bródka, P.: A method for group extraction and analysis in multilayer social networks. CoRR abs/1612.02377 (2016)
3. Chen, P., III, A.O.H.: Multilayer spectral graph clustering via convex layer aggregation: theory and algorithms. *IEEE Trans. Sig. Inf. Proc. Netw.* **3**(3), 553–567 (2017)
4. Dong, X., Nefedov, N.: Clustering on multi-layer graphs via subspace analysis on Grassmann manifolds. *IEEE Trans. Sig. Process.* **62**(4), 905–918 (2014)
5. Gao, C., Liang, M., Wang, Z., Zhou, Z.: Network community detection based on the Physarum-inspired computational framework. *IEEE ACM Trans. Comput. Biol. Bioinform.* **15**(6), 1916–1928 (2018)
6. Gao, C., Liu, J., Kurths, J.: Even central users do not always drive information diffusion. *Commun. ACM* **62**(2), 61–67 (2019)
7. Gligorićević, V., Zafeiriou, S.: Non-negative matrix factorizations for multiplex network analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(4), 928–940 (2019)
8. Leskovec, J., Lang, K.J., Mahoney, M.W.: Empirical comparison of algorithms for network community detection. In: The 19th International Conference on World Wide Web, Raleigh, pp. 631–640. ACM (2010)
9. Liu, W., Wang, S., Gong, M., Zhang, M.: An improved multiobjective evolutionary approach for community detection in multilayer networks. In: The 2017 IEEE Congress on Evolutionary Computation, Donostia, pp. 443–449. IEEE (2017)
10. Ma, X., Dong, D., Wang, Q.: Community detection in multi-layer networks using joint nonnegative matrix factorization. *IEEE Trans. Knowl. Data Eng.* **31**(2), 273–286 (2019)
11. Mucha, P.J., Richardson, T., Onnela, J.P.: Community structure in time-dependent, multiscale, and multiplex networks. *Science* **328**(5980), 876–878 (2010)
12. Newman, M.E., Girvan, M.: Finding and evaluating community structure in networks. *Phys. Rev. E* **69**(2), 026113 (2004)
13. Ni, J., Cheng, W., Fan, W., Zhang, X.: ComClus: a self-grouping framework for multi-network clustering. *IEEE Trans. Knowl. Data Eng.* **30**(3), 435–448 (2018)
14. Perozzi, B., Al-Rfou, R., Skiena, S.: DeepWalk: online learning of social representations. In: The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, pp. 701–710. ACM (2014)
15. Wang, H., Yang, Y., Liu, B.: GMC: graph-based multi-view clustering. *IEEE Trans. Knowl. Data Eng.* **32**(6), 1116–1129 (2020)
16. Xie, Y., Gong, M., Wang, S., Yu, B.: Community discovery in networks with deep sparse filtering. *Pattern Recognit.* **81**, 50–59 (2018)
17. Yang, L., Cao, X., Meng, D.: A unified semi-supervised community detection framework using latent space graph regularization. *IEEE Trans. Cybern.* **45**(11), 2585–2598 (2015)