Postprint: LPANNI Optimization Algorithm Based on Cycle Structure

Authors: Liu Ji, Jia Fangdi

Date: 2022-05-10T11:22:58+00:00

Abstract

To improve the accuracy of overlapping community detection, we propose CLPANNI (Cycle Label Propagation Algorithm with Neighbor Node Influence), an optimized algorithm based on cycle structures derived from LPANNI. The algorithm extracts minimal cycle information of nodes, measures node importance using the cycle ratio metric, and performs label updates in ascending order, thereby enhancing the stability of the label propagation process. It receives labels from neighboring nodes weighted by their influence. Compared with four baseline algorithms using the NMI_{LFK}, NMI_{MGH}, and MOV metrics, CLPANNI exhibits superior performance in community detection accuracy. Experimental results demonstrate that the algorithm can effectively detect overlapping community structures in networks, identify dense subgraphs, and produce community distributions that more closely approximate the true network structure.

Full Text

Preamble

LPANNI Optimization Algorithm Based on Cycle Structure

Liu Jia,b, **Jia Fangdi**a
(aSchool of Statistics & Data Science; bXinjiang Social & Economic Statistics & Big Data Application Research Center, Xinjiang University of Finance and Economics, Urumqi 830012, China)

Abstract: To improve the accuracy of overlapping community detection, this paper proposes CLPANNI (Cycle Label Propagation Algorithm with Neighbor

Node Influence), an optimization of the LPANNI algorithm based on cycle structure. The algorithm mines minimal cycle information of nodes, measures node importance using the cycle ratio index, and performs label updates in ascending order of importance, thereby increasing the stability of the label propagation process. Labels from neighbor nodes are weighted and received according to their influence magnitude. Compared with four benchmark algorithms across NMI_{LFK}, NMI_{MGH}, and Mov metrics, CLPANNI demonstrates superior performance in community detection accuracy. Experimental results show that the algorithm can effectively detect overlapping community structures in networks, identify tightly-knit subgroups, and produce community distributions that more closely approximate real network structures.

---

0 Introduction

Complex networks serve as abstract models for understanding real-world complex systems, where entities are represented as nodes and their relationships as edges. Community structure represents one of the most universal and important topological characteristics of complex networks, characterized by dense internal connections within communities and sparse connections between them. Community detection is crucial for exploring the operational mechanisms and functional properties of complex systems. Based on whether multiple community memberships for nodes are considered, community detection algorithms can be divided into two categories: non-overlapping community detection and overlapping community detection. In real networks, community structures typically exhibit overlapping regions, often containing overlapping nodes that simultaneously belong to two or more communities. These overlapping nodes play a vital role in network structure evolution, and their function in network dynamics warrants in-depth investigation in the era of big data characterized by universal connectivity.

Since 2005, scholars have attempted to design overlapping community detection algorithms from various perspectives to improve identification accuracy and computational efficiency. Among these, the label propagation algorithm has been widely applied to real-world network studies due to its linear time complexity advantage. COPRA (Community Overlap PRopagation Algorithm) was the first label propagation algorithm applied to overlapping community detection, identifying communities based on changes in nodes' affiliation coefficients across different communities. Subsequent algorithms such as SLPA (Speaker-Listener Label Propagation Algorithm), DEMON (Democratic Estimate of the Modular Organization of a Network), and ACSLPA (Active Semi-supervised SLPA) were proposed. Vinícius da Fonseca Vieira et al. conducted comparative studies of five classical algorithms—CPM (Clique Percolation Method), COPRA, DEMON,

SLPA, and BigCLAM (CLuster Affiliation Model for Big networks)—and found that algorithmically identified communities merely represent computational results rather than ground-truth communities. They noted that evaluating algorithm performance based solely on common metrics is problematic, suggesting that algorithm design should pay more attention to the number of overlapping nodes and their membership degrees.

Notably, most previous research has primarily focused on node adjacency relationships to study network functions and characteristics. However, real interaction scenarios often involve complex multi-node interactions. Whether in natural systems or virtual social networks, individual behavior is often associated with group dynamics. To achieve better coordination, individuals must consider not only pairwise interactions but also group-level interactions. Given the impact of feedback mechanisms on real network dynamics, particularly the positive and negative feedback effects of important overlapping nodes across different communities, a new perspective is needed to analyze node influence.

1.1 Overlapping Community Detection

Overlapping nodes are key elements affecting network topology evolution, making the exploration of their relationship with network dynamics and structure crucial. Overlapping community detection algorithms thus serve as effective methods for studying network dynamics. This paper focuses on algorithms based on clique percolation and label propagation ideas, where nodes aggregate into groups following bottom-up propagation rules. Unlike top-down algorithms such as modularity optimization, these approaches impose no special constraints on community structure quality, making them more consistent with spontaneously formed organizations or clusters in the real world. As network science enters a new research phase, higher-order interaction dynamics has attracted great interest, presenting challenges for overlapping community detection that must consider multi-node interaction features and mesoscale neighborhood information.

1.2 Related Work

Drawing on Poincaré's "subdivision" concept, references [15, 16] decomposed networks into homogeneous subnetworks, establishing a new framework for network science research: cycle structure. Since cycle structure bridges individual and group relationships, considering multi-node interactions to some extent, it can reflect nodes' local influence and further guide community detection. Reference [17] designed a new node importance index based on first-order cycle structure: the cycle ratio. Nodes identified by this metric are relatively dispersed, enabling efficient, non-redundant propagation with strong synchronization capability. Reference [18] proposed the LPANNI overlapping community detection algorithm, which combines advantages of COPRA and DLPA (Dominant Label Propagation Algorithm), cleverly solving COPRA's parameter determination

problem across different networks. LPANNI fully utilizes local node information by integrating node importance, neighborhood similarity, and neighbor node influence to reduce label propagation randomness. It also introduces a historical label preference strategy to determine each node's dominant label per iteration, improving overlapping community identification accuracy.

1.3 Evaluation Metrics

When ground-truth community structures are unknown, quality functions are generally used to measure community tightness, commonly including EQ and Qov. This paper adopts the Mov metric, which calculates each node's contribution to communities based on its affiliation strength across different communities —a precise overlapping measurement approach that aligns well with reference [13]'s recommendations. Mov measures node contribution based on the difference between intra-community and inter-community edge counts, effectively avoiding the resolution limit problem of other overlapping modularity metrics for highly overlapping community structures. The specific formula is:

$$M_{ov} = \sum_{c=1}^{K} \left[\frac{n_c^{in} - n_c^{out}}{n_c^{in} + n_c^{out}} \right] \cdot \left[\frac{1}{n} \sum_{i \in c} a_{ci} \right]$$

where n_c^{in} and n_c^{out} represent the number of nodes and edges in the r-th community c, respectively. Since the first factor ranges between -1 and 1 and the second factor between 0 and 1, Mov values vary between -1 and 1.

When ground-truth community structures are known, NMI metrics are commonly used to evaluate partitioning results. This paper employs two NMI metrics from the CDlib library (Community Discovery Library).

2.1 Symbol Description

Key symbols and their meanings are shown in Table 1.

Table 1. Symbol Description

Symbol	Meaning
NNI	Neighbor Node Influence
LNg	Node's dominant label set
Ng(i)	Number of overlapping nodes
	L'
b(c,i)	Node i's membership coefficient in community C
l(Cv,bv)	Node i's dominant label
T	Maximum iteration count

Symbol	Meaning

These symbols are primarily based on the proposed CLPANNI algorithm, with some parameters appearing in subsequent experiments.

2.2 Parameter Initialization

- Maximum iteration count T
- Node count V
- Iteration time t
- Node i' s membership strength across different communities represented by ordered pairs: $b_c(c, i)$
- Node i' s neighbor nodes $NG(i)$
- Node i' s dominant label D_i (the label with maximum membership coefficient in its label set)
- Node i' s label set size $|L'|$
- Node i' s label set L_i

Initially, each node in network $G(V, E)$ forms an independent community with membership coefficient 1, i.e., community i' s membership coefficient b_i is 1, denoted as $b_i(i, 1)$.

2.3 Update Strategy

Input: $G = (V, E, w)$, maximum iteration count T
Output: Community identification results

Phase 1: Fixed Label Update Order
The algorithm first calculates neighbor node importance according to the neighbor node influence formula for all nodes in V.

Phase 2: Label Propagation Process
At iteration $t = 0$, each node is initialized with $l[i] = \{i, 1\}$. The algorithm then iterates while $t < T$:

1. For each node, determine its dominant label D_i after the current iteration
2. If all nodes' label set sizes and dominant labels remain unchanged, the algorithm terminates
3. Otherwise, continue iterating until T is reached

Output: Label set L_i for each node i ($i \in V$)

The LPANNI algorithm consists of two phases: fixed label update order and label propagation. Phase 1 updates labels according to neighbor node influence magnitude, while Phase 2 determines dominant labels by propagating the

community label with maximum membership coefficient. When multiple labels share the same maximum membership coefficient, the algorithm prioritizes the dominant label from the previous iteration; otherwise, it selects one randomly.

LPANNI considers local node information and cleverly designs label update rules through neighbor node influence calculation, offering valuable insights for propagation strategies. However, it only focuses on pairwise node interactions, primarily utilizing node degree information, which provides limited discrimination between nodes with identical local structures and relies heavily on node ID ordering. This paper improves upon these strengths and weaknesses.

3 CLPANNI Algorithm Design

Complex interactions enable organizational growth and evolution, with inter-organizational interactions driving connectivity and development. From a network science perspective, specific structures emerge such as star structures, chain structures, and cycle structures. Cycle structure is a fundamental network component and one of the most important mechanisms for network functionality: the structural basis of feedback effects, which are critical to system evolution.

In network dynamics synchronization research, Shi Dinghua et al. discovered that networks with optimal synchronizability are nearly homogeneous networks with identical degrees, minimal path sums, and maximal girth. Cycles provide redundant connectivity paths structurally and represent feedback mechanisms functionally, creating reinforcement effects in network dynamics that enhance social coordination. Consequently, cycle structures are important for maintaining network connectivity and dynamic interactions. Building on this, Fan Tianlong et al. argued that nodes participating in many cycles are crucial, exerting significant influence on network connectivity, synchronization, and control. They designed a cycle structure-based node importance metric: the cycle ratio. Figure 1 illustrates the calculation of node 1' s cycle ratio in subgraph (a), while subgraph (c) shows degree, H-index, core number, cycle ratio, betweenness centrality, and other information for all nodes. They defined a new matrix for minimal cycles in first-order cycle structures called the cycle number matrix. This $n \times n$ matrix (where n is the number of nodes) describes co-cycle relationships between nodes, with matrix elements representing the number of cycles shared by any two nodes. Node i' s cycle ratio is then calculated as the sum of ratios between non-diagonal elements and diagonal elements in row i of the cycle number matrix.

The redundant connectivity and feedback mechanisms provided by cycles increase the probability of contact and synchronization for nodes on cycles during both synchronization and propagation processes, better simulating social reinforcement effects. Given the cycle ratio's outstanding performance in identifying high-propagation nodes, this paper applies it to label propagation algorithms, first using cycle ratio to locate nodes with strong propagation capabilities, then employing LPANNI' s label propagation strategy to improve community parti-

tioning accuracy.

3.1 Related Definitions

Simple Undirected Network $G(V, E)$: Where V and E represent node sets and edge sets respectively, with n nodes and m edges.

Cycle: A closed path with identical start and end points in a two-dimensional plane. Cycle size equals its number of edges; a minimal cycle is the smallest loop containing a given node.

Girth: The number of edges in the shortest loop starting and ending at a node, i.e., the length of the minimal cycle passing through the node.

Cycle Number: The quantity of minimal cycles containing a node.

Cycle Ratio (CR): A node importance metric calculated as the sum of ratios between elements in row i of the cycle number matrix and the diagonal element. The specific formula is:

$$CR_i = \begin{cases} 0.1, & \text{if } c_{ii} = 0 \\ \sum_{j \neq i, c_{ij} > 0} \frac{c_{ij}}{c_{ii}}, & \text{if } c_{ii} > 0 \end{cases}$$

where c_{ij} is the number of shared cycles between nodes i and j $(i \neq j)$ in matrix S. If $i = j$, c_{ii} represents the number of cycles containing node i. To precisely measure neighbor node influence magnitude, the first case is set to 0.1.

Node Similarity (Sim): This paper measures similarity based on network structure. Reference [25] surveyed local similarity metrics and analyzed their design principles, categorizing structural similarity indices into three types: local information-based, path-based, and random walk-based. The similarity metric from reference [18] combines local node information and path length, effectively integrating their advantages, and is therefore adopted here:

$$Sim(i, j) = \sum_{p \in P_{ij}} \frac{|A|^{|p|}}{|p|}$$

where p represents a path directly or indirectly connecting nodes i and j, $|p|$ denotes path length (varying from 1 to a), and $|A|^{|p|}$ measures path p. The path length threshold a controls computational complexity and distinguishes similarity differences arising from degree variations between nodes.

Neighbor Node Influence (NNI): Considering that neighbor nodes exert different influences due to their varying local structures, this paper adopts the NNI metric from reference [18], which objectively integrates neighbor node importance and similarity:

$$NNI(x) = \max_{y \in NG(x)} [CR(y) \cdot Sim(x,y)]$$

3.2 Improvements to LPANNI

LPANNI measures node importance and neighbor node influence using only node degree and triangle information, without considering additional cycle structures. This limited perspective provides insufficient discrimination between nodes with identical local structures. The cycle ratio metric identifies important nodes by measuring their participation in neighbor cycles, which benefits label propagation dynamics. This paper proposes CLPANNI, an overlapping community detection algorithm for undirected, unweighted networks that incorporates cycle structure information to fix label update order in ascending cycle ratio, thereby improving community identification accuracy.

3.3 CLPANNI Algorithm Framework

CLPANNI consists of two phases: (1) calculating node cycle ratios and neighbor node influences, and (2) performing label propagation to determine all nodes' community memberships and output their label sets. Figure 2 illustrates the specific steps, with the left portion showing Phase 1 and the right portion showing Phase 2.

Algorithm Flow: 1. Input graph $G(V, E)$ and maximum iteration count T 2. Calculate node cycle ratios 3. Calculate node similarities 4. Calculate node influences 5. Sort nodes in ascending importance order to obtain VQ 6. Update labels according to rules and normalize label membership coefficients 7. Repeat until node label sets stabilize or maximum iterations reached 8. Output node label sets, grouping nodes with identical labels into the same community

Previous algorithms [18, 26, 27] considered network triangle information. Visualizing cycle structure distribution in LFR data helps understand network cycle information. While detecting community structures using CLPANNI, the distribution of minimal cycles becomes clearly visible, as shown in Figure 3.

Table 2. Specific Parameters of Synthetic Network

Parameter	Value
On	
Om	

The six LFR networks in our experiments contain abundant cycle structures. Different scales exhibit distinct cycle distributions, with minimal cycle girth ranging from 3 to 8, and triangles comprising over half of all minimal cycles.

When the mixing parameter $\mu = 0.1$, triangles account for at least 90% of minimal cycles. When $\mu = 0.3$, as network structure clarity decreases, the total number of minimal cycles declines significantly, with triangle proportion decreasing while quadrilaterals and pentagons increase, partially explaining why community identification degrades when network structures become unclear. For fixed network scales, increasing topological complexity diversifies minimal cycle types and distributions, raising community detection difficulty.

3.4 Label Propagation Rules

The COPRA algorithm considers nodes' membership strength across communities, analogous to how individuals distribute attention or energy across different aspects (summing to 1). LPANNI increased algorithm stability by fixing label propagation order through ascending node importance and solved COPRA's parameter tuning problem through neighbor influence-based update strategies and historical label preferences. This paper adopts a cycle perspective for measuring node importance:

Phase 1: Sort nodes by cycle ratio in ascending order. For nodes with identical cycle ratios, sort by ID ascending to obtain fixed update sequence VQ.

Phase 2: Update labels according to VQ order. Initially, each node fully belongs to its own community: $L_{di} = (i, 1)$. Then, following VQ order, nodes receive dominant labels from neighbors to form label set LNg, where dominant labels refer to the maximum membership coefficient and community label transmitted by neighbor nodes.

Membership coefficients are weighted by neighbor node influence NNI to produce new label set L' (size equals neighbor count). Each node's total membership coefficient becomes 1. After adaptive removal of useless labels and normalization, the iteration's final label set L'' is obtained. After T iterations, each node's label set is output.

The label with maximum membership coefficient is identified as node i's dominant label. If multiple dominant labels exist, the previous iteration's dominant label is selected; otherwise, one is chosen randomly. Iteration stops when all nodes' label sets and dominant labels stabilize, outputting the final label sets.

4.1 Experimental Data

1) Artificial LFR Benchmark Networks
LFR benchmark networks generate synthetic data approximating real networks where both node degrees and community sizes follow power-law distributions. We generated two groups with mixing parameters $\mu = 0.1$ and $\mu = 0.3$. Each group contains three scales (1,000, 3,000, and 5,000 nodes), with five different overlapping degrees per scale (details in Table 2).

2) Minimal Cycles in Synthetic Networks

Previous overlapping community detection algorithms rarely examined network cycle structure distributions. Visualizing LFR data cycle distributions aids understanding. While CLPANNI detects community structures, minimal cycle distributions become apparent (Figure 3).

3) Real-world Datasets

Reference [12] provides three real networks with known community structures: a co-purchasing network (Amazon), a scientific collaboration network (DBLP), and a friendship network (YouTube), including overlapping node information. Minimal cycle mining revealed that Amazon and DBLP contain few cycle types, while YouTube exhibits diverse cycle distributions with girth ranging from 3 to 10, indicating relatively complex structure (Table 3).

Table 3. Specific Information about Real Networks

Network	Max Community Size	Min Community Size	Overlapping Nodes (%)	Minimal Cycle Count	Minimal Cycle Distribution
Amazon			1,394 (18%)		{(3,26061),(4,110)}
DBLP			214 (3.3%)		{(3,108880),(6,2)}
YouTube			865 (13%)		{3,4,5,6,7,8,9,10}

4.2 Experimental Results

Using SLPA, DEMON, CPM, and LPANNI as baseline algorithms, parameters were tuned through multiple experiments (Table 4). Due to SLPA's instability, results were averaged over 10 repetitions. For CPM, parameter k was tested from 3 to 6, with $k = 3$ yielding optimal results. Other parameters used CDlib library defaults.

Table 4. Specific Parameters of Benchmark Algorithms

Algorithm	Amazon	YouTube	Parameters
DEMON	min_{com}_{size}=3, epsilon=0.25		
LPANNI	T=21, r=0.1	T=21, r=0.2	
CLPANNI	T=20, a=3, b=0	T=20, a=3, b=0	

LFR benchmark tests used NMI_{LFK} and NMI_{MGH} metrics. NMI_{LFK} extends normalized mutual information for overlapping communities but may overestimate similarity. NMI_{MGH} (also called NMI-max) optimizes NMI_{LFK} and was adopted for our tests.

Figure 5 shows CLPANNI and LPANNI outperform other baselines, accurately identifying community structures even as network scale increases and community clarity decreases. With more overlapping nodes, CLPANNI slightly outperforms LPANNI, demonstrating that cycle ratio better captures node importance.

Real network tests (Figure 6) confirm CLPANNI's superiority over LPANNI across most metrics except YouTube, where performance is slightly lower. On the Mov metric, CLPANNI shows clear advantages, particularly on Amazon data. SLPA performs relatively well on DBLP, indicating effectiveness on networks with clear community structures, but suffers from extreme instability and randomness, requiring multiple repetitions for reliable results. DEMON's low NMI_{MGH} scores on YouTube suggest democratic voting mechanisms are unsuitable for such networks. Overall, CLPANNI demonstrates robust performance across multiple metrics.

5.1 Algorithm Detection Results

Experimental results (Figure 5) show CLPANNI performs increasingly better relative to other algorithms as community structures become fuzzier and overlap increases. On DBLP and Amazon networks, CLPANNI improves upon the original algorithm (Figure 6). Multiple algorithms show suboptimal performance on YouTube data, necessitating deeper analysis. As social network data, YouTube exhibits social network reinforcement effects. (Note: Due to SLPA's randomness, 15 repetitions were performed, showing median results of 477 detected communities.)

In reality, YouTube's 6,426 nodes contain 865 overlapping nodes across 1,078 communities, with the dominant community size being 5 (307 communities). Table 5 presents each algorithm's performance on YouTube, including overlapping modularity, dominant community sizes, detection counts, and total node frequencies. SLPA detected 455 communities. CPM ($K = 3$) identified 230 triangle-based cliques. DEMON found the fewest communities with highest Mov score but exhibited overfitting in overlapping node detection. CPM and SLPA identified similar community counts (455). LPANNI discovered the most communities but they were smaller. While CLPANNI's overlapping node detection is less effective than LPANNI's, it shows higher accuracy in identifying dominant community sizes, indicating better fit for community scale detection. YouTube's diverse minimal cycle distribution suggests CLPANNI performs better on networks with redundant edge structures and complex cycles, better simulating social reinforcement effects.

Table 5. Algorithmic Detection Results

Algorithm	Mov Score	Dominant Community Size & Count	Communities Detected	Total Node Frequency
DEMON		{5,307}		
LPANNI		{3,230}, {5,120}, {4,46}		
CLPANNI		{2,167}, {5,129}		

5.2 Community Size Distribution

LPANNI, CLPANNI, DEMON, and SLPA all incorporate label propagation concepts and thus share common drawbacks: label over-propagation and large communities absorbing small ones. After removing outliers, community detection results are shown in Figure 7. CPM identifies community structures through clique percolation, so community sizes strongly correlate with local structural density. In real YouTube data, the largest community contains 31 nodes, yet all five algorithms identified larger maximum communities, indicating tightly connected components exist. These results were obtained under CLPANNI' s most relaxed conditions without targeted parameter tuning; careful selection of node membership thresholds could further improve detection but would consume more time.

5.3 Visualization Analysis

Figure 8 shows Louvain algorithm results from Gephi, where same-colored nodes belong to the same community and labeled nodes are overlapping nodes spanning at least 8 communities. Visualization reveals YouTube's unclear community structure with massive connected components and concentrated high-overlap node distributions, partially explaining suboptimal algorithm performance. Due to concentrated overlapping nodes, YouTube facilitates large-scale opinion formation during information propagation, suggesting that monitoring overlapping node opinions should be prioritized for network public opinion early warning and guidance.

6 Conclusion

This paper optimizes LPANNI by proposing CLPANNI, a cycle structure-based label propagation algorithm for overlapping community detection that improves identification accuracy for tightly-knit network structures. CLPANNI offers good stability and simultaneously obtains each node's minimal cycle information and the distribution of minimal cycles with different girths across the network. This helps understand network structural characteristics and provides mesoscale information to guide overlapping community detection.

Future work will combine cycle structures for deeper analysis of networks with dense structures. Real networks often evolve dynamically with environmental changes, while topology affects information transmission. Utilizing detected overlapping nodes and community structures to analyze network dynamic evolution characteristics warrants further research. Given that network connections often represent limited choices under incomplete information, future studies should integrate prior knowledge, incorporate high-order network information to quantify networks more reasonably, and nest node attribute information to mine overlapping community structures and identify key overlapping nodes to help predict network dynamic evolution directions.

References

[1] Palla G, Deranyi I, Farkas I, et al. Uncovering the overlapping community structure of complex networks in nature and society [J]. Nature, 2005, 435(7043): 814.

[2] Li Hui, Chen Fucai, Zhang Jianpeng, et al. Survey of community detection algorithms in complex network [J]. Application Research of Computers, 2021, 38(06): 1611-1618. DOI: 10.19734/j.issn.1001-3695.2020.06.0211.

[3] Xu Ying. Novel algorithm of overlapping community detection and analysis with improved ant colony algorithm [J]. Application Research of Computers, 2020, 37(05): 1375-1379. DOI: 10.19734/j.issn.1001-3695.2018.10.0803.

[4] Fu Rao, Meng Fanrong, Xing Yan. Overlapping Community Discovery Algorithm Based on Node Importance and Similarity [J]. Computer Engineering, 2018, 44(9): 192-198.

[5] Chu Yangjie, Yang Zhongbao, Hong Ye. Local extension approach through genetic algorithm for overlapping community detection [J]. Application Research of Computers, 2019, 36(04): 1106-1109.

[6] Shchur O, S Günnemann. Overlapping Community Detection with Graph Neural Networks [J]. https://arxiv.org/abs/1909.12201v1

[7] Raghavan U N, Albert R, Kumara S. Near Linear Time Algorithm to Detect Community Structures in Large-Scale Networks [J]. Physical Review E, 2007, 76(3 Pt 2): 036106.

[8] Zhang Ying-long, Xia Xuewen, Xu Xing, et al. Review on label propagation algorithms for community detection [J]. Journal of Chinese Computer Systems, 2021, 42(5): 1093-1102.

[9] Gregory S. Finding overlapping communities in networks by label propagation [J]. New Journal of Physics, 2009, 12(10): 2011-2024.

[10] Xie J, Szymanski B K, Liu X. SLPA: Uncovering Overlapping Communities in Social Networks via A Speaker-listener Interaction Dynamic Process [J]. IEEE, 2012.

[11] Coscia M, Rossetti G, Giannotti F, et al. DEMON: a Local-First Discovery Method for Overlapping Communities [C]//Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012.

[12] Alghamdi E, Greene D. Active semi-supervised overlapping community finding with pairwise constraints [J]. Applied Network Science, 2019, 4(1): 1-27.

[13] Vieira V F, Xavier C R, Evsukoff A G. A comparative study of overlapping community detection methods from the perspective of the structural properties [J]. Applied Network Science, 2020, 5(1): 1-42.

[14] Yang J, Leskovec J. Structure and overlaps of ground-truth communities in networks [J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2014, 5(2): 1-35.

[15] Shi D, Chen G. Searching for Optimal Network Topology with Best Possible Synchronizability [J]. Circuits and Systems Magazine, IEEE, 2013, 13(1): 66-75.

[16] Shi D, Linyuan L, Chen G. Totally Homogeneous Networks [J]. arXiv: 1309.6310.

[17] Fan T, Lü L, Shi D, et al. Characterizing cycle structure in complex networks. Commun Phys 4, 272 (2021). https://doi.org/10.1038/s42005-021-00781-3

[18] Lu M, Zhang Z, Qu Z, et al. LPANNI: Overlapping Community Detection Using Label Propagation in Large-Scale Complex Networks [J]. IEEE Transactions on Knowledge and Data Engineering, 2019, 31(9): 1736-1749.

[19] He-Li Sun, Jian-Bin, et al. Detecting overlapping communities in networks via dominant label propagation [J]. Chinese Physics B, 2015, 24(1): 24 018703.

[20] Shen H W. Detecting the Overlapping and Hierarchical Community Structure in Networks [J]. Springer Berlin Heidelberg, 2013.

[21] Nicosia V, Mangioni G, Carchiolo V, et al. Extending the definition of modularity to directed graphs with overlapping communities [J]. Journal of Statistical Mechanics Theory & Experiment, 2009, 2009(03): 3166-3185.

[22] Lázár A, Ábel D, Vicsek T. Modularity measure of networks with overlapping communities [J]. EPL (Europhysics Letters), 2010, 90(1): 18001.

[23] Rossetti G, Milli L, Cazabet R. CDLIB: a python library to extract, compare and evaluate communities from complex networks [J]. Applied Network Science, 2019, 4(1).

[24] Fan Tianlong. Cycle structure in a network [EB/OL]. https://swarma.org/?p=31541. 2021-8-10/2022-4-4.

[25] Li Yan-li, Zhou Tao. Local Similarity Indices in Link Prediction [J]. Journal of University of Electronic Science and Technology of China, 2021, 50(03): 422-427.

[26] Zheng Wenping, Bi Xinqi, Yang Gui. An overlapping community detection algorithm based on asymmetric triangle cuts [J]. Journal of Nanjing Normal University (Engineering and Technology Edition), 2022, 22(01): 1-8.

[27] Pan Jian-fei, Dong Yi-hong, Chen Hua-hui, et al. The Overlapping Community Discovery Algorithm Based on Compact Structure [J]. Acta Electronica Sinica, 2019, 47(01): 145-152.

[28] Lancichinetti A, Fortunato S, J Kertész. Detecting the overlapping and hierarchical community structure of complex networks [J]. New Journal of Physics, 2009, 11: 033015.

[29] Mcdaid A F, Greene D, Hurley N. Normalized Mutual Information to evaluate overlapping community finding algorithms [J]. arXiv: 1110.2515 (2011).

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv −Machine translation. Verify with original.