



Sensor Network Structure Recognition Based on P-law

Chuiju You¹, Guanjun Lin^{1,*}, Jinming Qiu¹, Ning Cao¹, Yundong Sun² and Russell Higgs³

¹School of Information Engineering, Sanming University, Sanming, 365004, China

²School of Astronautics, Harbin Institute of Technology, Harbin, 150001, China

³School of Mathematics and Statistics, University College Dublin, Dublin, Dublin 4, Ireland

*Corresponding Author: Guanjun Lin. Email: guanjun.lin@fjismu.edu.cn

Received: 16 December 2021; Accepted: 24 January 2022

Abstract: A sensor graph network is a sensor network model organized according to graph network structure. Structural unit and signal propagation of core nodes are the basic characteristics of sensor graph networks. In sensor networks, network structure recognition is the basis for accurate identification and effective prediction and control of node states. Aiming at the problems of difficult global structure identification and poor interpretability in complex sensor graph networks, based on the characteristics of sensor networks, a method is proposed to firstly unitize the graph network structure and then expand the unit based on the signal transmission path of the core node. This method which builds on unit patulousness and core node signal propagation (called *p-law*) can rapidly and effectively achieve the global structure identification of a sensor graph network. Different from the traditional graph network structure recognition algorithms such as modularity maximization and spectral clustering, the proposed method reveals the natural evolution process and law of graph network subgroup generation. Experimental results confirm the effectiveness, accuracy and rationality of the proposed method and suggest that our method can be a new approach for graph network global structure recognition.

Keywords: Sensor network; graph network; P-law; unit subgroup; structure recognition

1 Introduction

The sensor graph network is a sensor network model organized according to the structure of the graph network (the sensors are the nodes of the graph network, the connections between the sensors serve as the edge of the graph network), and it is a distributed intelligent network system composed of sensor nodes interconnected through wireless communication deployed in the area of action, which can perform the specified tasks independently according to the environment. The self-organizing behavior of the sensor network is manifested as the clustering of sensor nodes in the sensor graph network. In the sensor graph network, there is a sensor network unit subgroup structure composed of several sensors with the core node (such as the gateway of the Internet of things (IoT)) as the center. These unit subgroups can be regarded as the basic organization module of a sensor network. The central node is the organizational



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

core of the unit subgroup and has an aggregation effect. For example, the IoT gateway generally undertakes the tasks of data exchange, storage, processing and so on. These unit subgroups can be further combined to form more complex and larger sensor network subgroups. The unit and centrality of the sensor network are identified as the core of sensor network subgroup structure.

As a type of graph network, sensor networks widely exist in the real world, such as meteorological monitoring station network, energy consumption monitoring network, urban brain infrastructure network and so on. Many problems of sensor networks are essentially graph network structure problems. For example, the urbanization control of energy consumption of buildings, control of electric energy between high-rise and low-rise buildings, which is a graph network structure problem; There are many resource scheduling problems in the infrastructure of urban brain, including hardware resource management, capturing resource bottlenecks, system changes of user dynamic graph, etc. These are structural problems such as graph decomposition and graph synthesis.

At present, the research on sensor networks involves sensor network routing protocol [1–3], sensor network security [4,5], internet of things resource scheduling [6,7], energy consumption management [8] and so on. However, there is little research on sensor network structure recognition. The research of sensor graph network structure and graph network structure comes down in one continuous line. In recent years, the research on the structure of graph network has attracted widespread attention. Many graph network structure recognition algorithms have been developed in many disciplines such as computer science, physics and mathematics, and are widely used in specific problems in various scientific fields [9,10]. The existing methods for graph network structure recognition mainly include modularity maximization algorithm [11,12], spectral clustering algorithm [13,14] and label propagation algorithm [15,16]. Due to the large scale of large-scale graph networks, it is difficult to find global structure information. Therefore, the discovery of local structure has attracted the attention of researchers. The basic idea is to start from a few known nodes in the target structure and gradually expand to the remaining nodes in the structure. Research shows that the method based on seed set expansion [17,18] can effectively find local structures, including local spectrum method [19], personalized PageRank method [20,21], thermonuclear diffusion method [22], etc. Considering the unit characteristics and the connection mode between units which are based on core node signal transmission in sensor graph networks, a method based on unit patulousness and core node signal propagation (called p -law) is proposed to achieve the global structure recognition of sensor graph networks. This method first decomposes the sensor graph network according to the graph network units, and then combines the graph network units by the signal transmission path of the core node to form the sensor graph network subgroup. Specifically, the ideas of the proposed method are as follows: take the graph network unit subgroup as the basic construction module of the graph network subgroup. Then, take the central node as the signal organization center of the subgroup structure, and quickly generate the subgroup by the unit expansion method.

In summary, the main contributions of this paper are as follows:

- (1) The concepts of unit subgroup and k -order subgroup are provided, and the k -order subgroup model of graph network is established.
- (2) A k -order subgroup structure recognition method of graph network based on p -law is proposed.
- (3) The typical application of graph network subgroup structure identification method in sensor networks is presented.

2 Graph Network K-order Subgroup Model

2.1 K-order Subgroups of Graph Networks

Convention: let $G = (V, E)$ be a graph, where $V = \{v_1, v_2, \dots, v_n\}$ is the node-set of graph G , and

$E = \{e_1, e_2, \dots, e_m\}$ is the set of edges of graph G , and $j(v_i, v_j)$ is the order (HOPS) of node v_i reaching node v_j , and $deg(v)$ is the degree of node v (the number of edges connected to node v).

Definition 1: 1-order unit subgroup

Let $H = (V', E')$ be a subgraph of $G = (V, E)$, having $V' \subseteq V$, $E' \subseteq E$ and nodes $v \in V$, so that given any other nodes v'' in H , $j(v'', v) = 1$ is satisfied. Then H is the unit subgroup with node v as the core in graph G . which is recorded as:

$$g^{(v,1)} = g(G|H, v, k = 1) \tag{1}$$

In Eq. (1), v is the core of the 1-order unit subgroup g in the graph network G , and k is the order of the subgroup in the graph network G .

Definition 2: 1-order extension of subgroups:

If $g^{(v,1)} = g(G|H, v, 1)$ is a unit subgroup with v as the core, $g_i^{(v_i,1)} = g_i(G|H_i, v_i, 1)$ is the i -th unit subgroup of the graph network G with v_i as the core, $H = (V', E')$, $H_i = (V'_i, E'_i)$, $v' \in V'$, $v'_i \in V'_i$. If $v'_i \in V'$, $v'_i \neq v'$, then $g_i^{(v_i,1)}$ is the 1-order extension of $g^{(v,1)}$ and v'_i is the extended core of $g^{(v,1)}$. The expanded subgroup is a 2-order subgroup, which is recorded as:

$$g^{(v,s_k,2)} = g^{(v,[v_1,v_2,\dots,v_i],2)} = g^{(v,1)} \cup (g_1^{(v_1,1)} \cup g_2^{(v_2,1)} \cup \dots \cup g_i^{(v_i,1)}) \tag{2}$$

In Eq. (2): $s_k = [v_1, v_2, \dots, v_i]$, v is the core of the graph network subgroup g , and s_k is the set of 1-order extended cores of the graph network subgroup g .

The unit subgroups of the graph network and the 1-order extension of subgroups defined in definitions 1 and 2 are shown in Fig. 1.

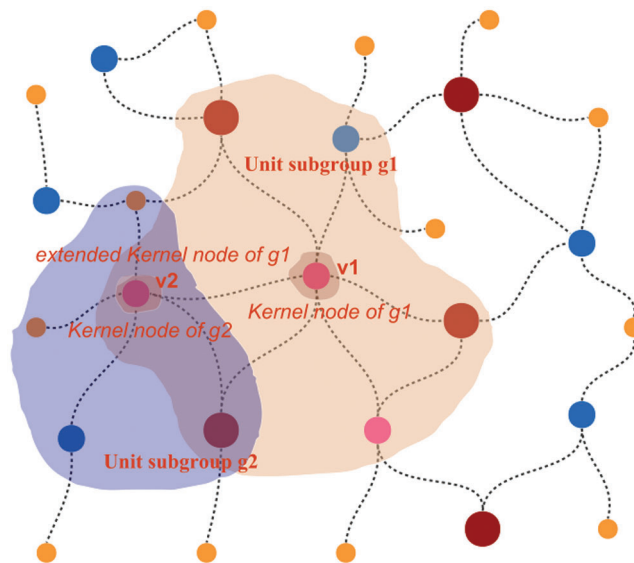


Figure 1: Schematic diagram of unit subgroups and 1-order extensions of graph network

Definition 3: k-order subgroup of graph network

if $g^{(v,s_k,1)}$ is a 1-order subgroup with v as the core, and $g^{(v,s_1,s_2,\dots,s_j,k)}$ satisfies: $g^{(v_{hi},s_{h+1},1)}$ is a 1-order subgroup with v_{hi} as the core, s_{h+1} is its 1-order extended core, where $v_{hi} \in s_h, h = 1, 2, \dots, k-2$, then, the following equation is called a k -order subgroup:

$$g^{(v,s_1,s_2,\dots,s_{k-1},k)} = g^{(v,s_1,1)} \cup g^{(v_{1i},s_2,1)} \cup \dots \cup g^{(v_{hi},s_{h+1},1)} \quad (3)$$

In Eq. (3), s_1, s_2, \dots, s_{k-1} is the set of k -th order extended cores of the graph network subgroup G .

Fig. 2 is a schematic diagram of the extension of k -order subgroup and k -order subgroup.

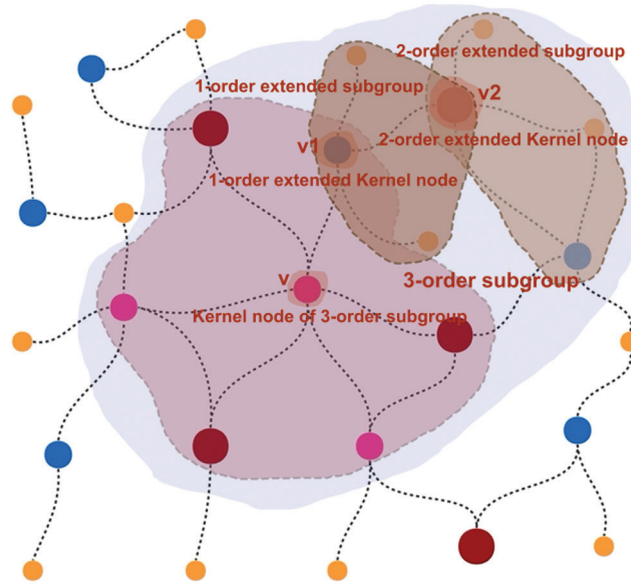


Figure 2: Schematic diagram of k -order subgroup extension and k -order subgroup

2.2 Related Properties of K-order Subgroup in Graph Network

Property 1: The k -order subgroup $g^{(v,s_1,s_2,\dots,s_{k-1},k)}$ of graph network has an extension chain $l(v, s_1, s_2, \dots, s_{k-1})$ with $v, s_1, s_2, \dots, s_{k-1}$ as the extension core series, and has a hierarchical structure corresponding to the extension chain $l(v, s_1, s_2, \dots, s_{k-1})$.

According to the definition of k -order subgroup, k -order subgroup expands gradually according to the core node series $v, s_1, s_2, \dots, s_{k-1}$, and the extension core expands step by step to form the extension chain $l(v, s_1, s_2, \dots, s_{k-1})$, and the extension forms k -order subgroup. Therefore, it has a k -layer hierarchy corresponding to the extension chain $l(v, s_1, s_2, \dots, s_{k-1})$.

Property 2: The order (hops) of any two nodes v_i and v_j in the k -order subgroup $g^{(v,s_1,s_2,\dots,s_{k-1},k)}$ of the graph network through the core v satisfies:

$$j_v(v_i, v_j) \leq 2k \quad (4)$$

Proof: According to property 1, the k -order subgroup of the graph network is hierarchical. If v_i and v_j are just on the k -th layer of subgroup g , at this time, the core v of the subgroup is the farthest from v_i and v_j , that is, $j(v_i, v) = k, j(v_j, v) = k$.

According to definition 3, v_i must reach v_j through the core v . At this time, if v_i and v_j are in layer K or within layer K , then: $j(v_i, v) \leq k, j(v_j, v) \leq k$. So, there is: $j_v(v_i, v_j) = j(v_i, v) + j(v_j, v) \leq 2k$.

End of proof.

Properties 1-2 are shown in Fig. 3.

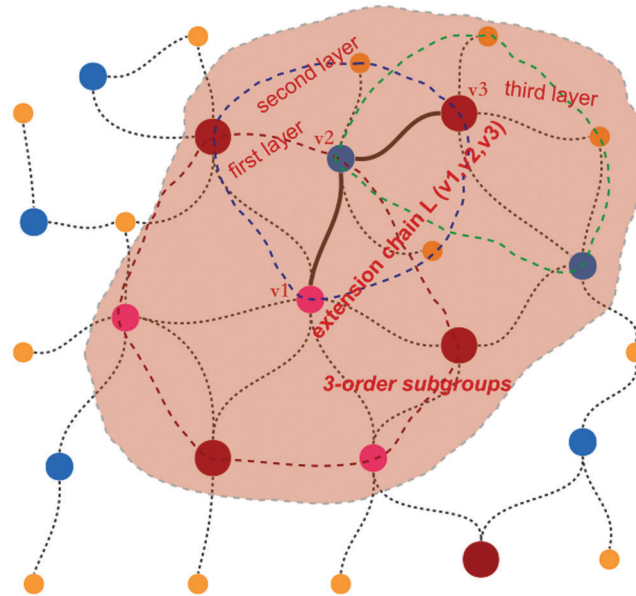


Figure 3: Schematic diagram of hierarchical structure of k-order subgroup

Theorem 1: Core reachability theorem of k-order subgroup of graph network

Any node v_i in k-order subgroup $g^{(v,s_1,s_2,\dots,s_{k-1},k)}$ of graph network can reach any other node v_j through the core node v .

Proof: According to definition 3 and property 1, the nodes of any layer in k-order subgroup of graph network are directly connected with the extension core of the layer. Because there is an extension chain in k-order subgroup of graph network, and the extended core on the extended chain is interconnected, any node in any layer of subgroup can reach the core v of the subgroup along the extended chain. Then, it reaches any other node in the subgroup through the extended chain. Therefore, any nodes in the subgroup can reach each other through the core and the extended chain, which has core reachability.

End of proof.

Fig. 4 shows the reachability of the core node of k-order subgroup. The node v_4 in the 3-order subgroup reaches the core v_1 through the extended chain $L (v_1, v_2, v_3)$, and then reaches the node v_5 through the core v_1 .

Theorem 2: Node reachability theorem between subgroups of graph network

Let $g_1^{(v,s_1,s_2,\dots,s_{k-1},k)} = g_1(G|H_1, v, s_1, s_2, \dots, s_{k-1}, k), g_2^{(v',s'_1,s'_2,\dots,s'_{l-1},l)} = g_2(G|H_2, v', s'_1, s'_2, \dots, s'_l, l)$ be two subgroups of graph network G respectively. If $H_1 \cap H_2 \neq \emptyset$, any node between $g_1^{(v,s_1,s_2,\dots,s_{k-1},k)}$ and $g_2^{(v',s'_1,s'_2,\dots,s'_{l-1},l)}$ can reach each other.

Proof:

According to the meaning, because $H_1 \cap H_2 \neq \emptyset, g_1^{(v,s_1,s_2,\dots,s_{k-1},k)}$ and $g_2^{(v',s'_1,s'_2,\dots,s'_{l-1},l)}$ must have a common node v'' .

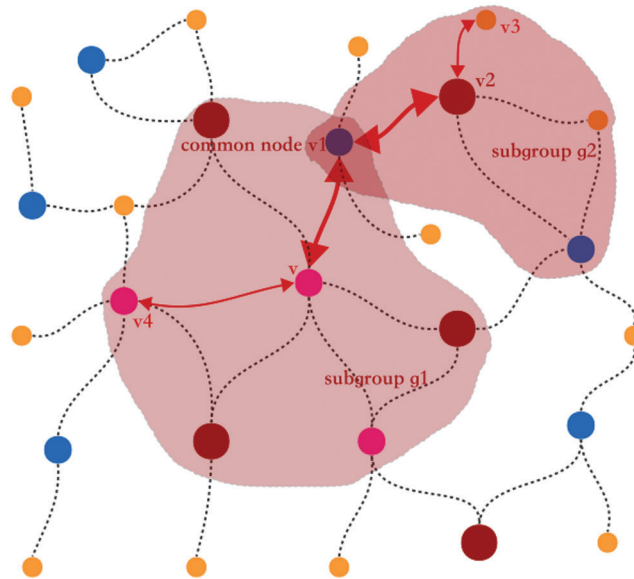


Figure 5: Schematic diagram of mutual accessibility of any nodes among subgroups

Contraction of k-order subgroup in graph network is by removing 1-order unit subgroup associated with the end extension core node in the extension chain (except the core node).

Proof:

Let $g^{(v,s_1,s_2,\dots,s_{k-1},k)}$ be a k-order subgroup and $l(v, s_1, s_2, \dots, s_{k-1})$ be its extended chain. Select the end extended core node s_{k-1} from $l(v, s_1, s_2, \dots, s_{k-1})$. Let $v_i \in s_{k-1}, g^{(v_i,1)}$ be a 1-order unit subgroup with v_i as the core. According to Theorem 3, if the subgroup is to be contracted, only remove the 1-order unit subgroup $g^{(v_i,1)}$ from $g^{(v,s_1,s_2,\dots,s_{k-1},k)}$ (except the core node v_i).

Namely: $(g^{(v,s_1,s_2,\dots,s_{k-1},k)} - g^{(s_{k-1},1)}) \cup s_{k-1}$. At this time, the k-order subgroup $g^{(v,s_1,s_2,\dots,s_{k-1},k)}$ shrinks to k-1-order subgroup $g^{(v,s_1,s_2,\dots,s_{k-2},k-1)}$ and the extension chain $l(v, s_1, s_2, \dots, s_{k-1})$ shrinks to $l(v, s_1, s_2, \dots, s_{k-2})$.

End of proof.

3 Recognition Algorithm for K-order Subgroup Structure of Graph Network

3.1 The Basic Flow of Subgroup Structure Recognition of Graph Network

The basic flow of k-order subgroup structure recognition algorithm of graph network is as follows:

- (1) Determine the order of subgroup structure. You can set the default value to 3. According to the general law, if the order is too large, the subgroup scale may be too large, the compactness of the subgroup structure will be reduced, and the control force of the core node of the subgroup structure on the subgroup will be weakened.
- (2) The unit subgroups are generated in the order of the degree of the nodes with which as core of unit subgroups in graph network (from high to low), until the generated unit subgroups contain all the nodes of the graph network.
- (3) Select the unit subgroup with high degree of core node from the unit subgroups in turn for k-order expansion until it cannot be extended.
- (4) If necessary, the generated k-order subgroups can be further synthesized and optimized.

3.2 Generation Algorithm for K-order Subgroups of Graph Network

Algorithm-1: Graph network subgroup generation algorithm

Input: graph $G(V, E)$, order of subgroups K

Output: k-order subgroups of graph network G

Process:

01 Generating a degree list of graph G nodes sorted by descending order:degree_list

02 Generating node_list according to the degree list

03 Generating unit subgroup of G :

04 init unit subgroup:unit_subgroup_dict = {key : value} //key

05 For a node in node_list

06 Select the Associated nodes of the node from G

07 Add node(core node)and the Associated nodes to *dict*, set the core node as key, set the Associated nodes as value

08 If unit_subgroup_dict includes all the nodes of G

09 return unit_subgroup_dict

10 end

11 Generating k-order subgroup:geSubGroup(G,K , unit_subgroup_dict) // K is the order of the subgroup

12 init expand_subgroup_list

13 while unit_subgroup_dict *dict* has more elements:

14 select the higher degree unit subgroup from unit_subgroup_dict as a subgroup to be extended

15 remove the unit subgroup from unit_subgroup_dict

16 for k in K : // Extend the k-order subgroups until they can no longer be extended

17 set Intersection of the **unit subgroup(exception the core of the unit subgroup)** and **keys of the unit_subgroup_dict** to intersection_node

18 for node in intersection_node

19 select the **unit subgroup by node(key) from unit_subgroup_dict, implements 1-order subgroup extension, set expanded subgroup to expand_subgroup**

20 end

21 end

22 add expand_subgroup to expand_subgroup_list

23 end

24 return expand_subgroup_list // Returns the generated k-order subgroup

3.3 Synthesis and Optimization Algorithm for K-order Subgroups of Graph Network

Algorithm-2: Synthesis and optimization algorithm of k-order subgroups in graph networks

Input: Graph $G(V,E)$, k -order-subgroups, K (similarity coefficient among subgroups)

Output: Subgroups of graph networks after synthesis and optimization

Process:

subgroup synthesis and optimization: synAndOptSubgroup (G , k -order-subgroups, K)

1 **init** Subgroups list of graph networks: subgroups_list = []

2 **init** dict_subgroups = []

3 **for** subgroup in k-order-subgroups:

4 dict_subgroups = [subgroup]

5 remove subgroup from k_order_subgroups

6 set nodes of subgroup to subgroup_1_nodes

7 **for** next_subgroup in k-order-subgroups:

8 set nodes of next_subgroup to subgroup_2_nodes

9 set intersection of subgroup_1_nodes and subgroup_2_nodes to intersection_1_2_subgroup

10 set size intersection_1_2_subgroup to size_intersection_1_2_subgroup

11 set minimum size of subgroup_1_nodes and subgroup_2_nodes to min_size_1_2_subgroup

12 set round (size_intersection_1_2_subgroup/min_size_1_2_subgroup, 2) to result

13 if result > k

14 dict_subgroups.append(next_subgroup)

15 remove next_subgroup from k_order_subgroups

16 **end**

17 subgroups_list.append(dict_subgroups)

18 **end**

19 **for** plus_subgroup in unit_subgroups(unit subgroups not composited in k-order-subgroups):

20 set nodes of plus_subgroup to nodes_plus_subgroup

21 **init** max_size_list

22 **for** subgroups in subgroups_list:

23 **init** nodes_list, result_nodes

24 **for** group in subgroups:

25 add nodes of group to nodes_list

26 **end**

27 **for** nodes in nodes_list:

28 add intersection of nodes and nodes_plus_subgroup to result_nodes

29 **end**

30 set max(result_nodes) to max_group

(Continued)

Algorithm 2: (continued).

```

31     add max_group to max_size_list
32     end
33     index =max_size_list.index(max(max_size_list))
34     add plus_subgroup to subgroups_list[index]
35 end
36 return subgroups_list

```

3.4 Analysis of Recognition Algorithm for K-order Subgroup Structure of Graph Network

The key points of recognition algorithm for k-order subgroup structure of graph network are as follows:

- (1) Unit subgroup generation: the 1-order unit subgroup generation method used in the algorithm is to sort the nodes from high to low according to the degree of nodes, and then take out the nodes in order to generate 1-order unit subgroup until the nodes contained in the generated 1-order unit subgroups cover all the nodes in the graph network. In the generation of 1-order unit subgroups, this method ensures the clustering centrality and cohesion of nodes, that is, the nodes with higher degree give priority to the generation of 1-order unit subgroups.
- (2) Generation of k -order subgroup: select the 1-order unit subgroup with the highest centrality and cohesion as the subgroup to be expanded firstly, and then select the corresponding unit subgroup from the rest 1-order unit subgroup to expand until it can no longer be expanded.
- (3) Synthesis and optimization of k -order subgroups: for the previously generated k -order subgroups, further synthesis and optimization can be carried out as needed. If two k -order subgroups have high similarity (a coefficient can be set here for similarity control), that is, there are more common nodes, they can be combined to ensure their cohesion.
- (4) After subgroups synthesis and optimization, there may be common nodes between subgroups, which are the channels to realize signal transmission between subgroups. If you want to define the subgroup attribution of these coincident nodes, the following principles can be adopted: 1) If the common node is the core node, it can be determined according to the subgroup attribution of the core node (the core node can only belong to one subgroup); 2) If the common node is not a core node, it can be determined by the number of associated edges of the node and each subgroup; 3) Adjustment of subgroup belonging of core nodes: the associated nodes of core nodes may belong to multiple subgroups, which can be determined by the number of associated nodes between the core node and each subgroup.

4 Experiments and Analysis**4.1 Data Description**

The following is the simulation experiment of sensor graph network structure recognition using the general real data set of graph networks: Dolphin network and karate club network. The two graph networks have 62 nodes, 128 edges, 34 nodes and 78 edges respectively. The real subgroup structure is two subgroups. The network diagram is shown in Fig. 6. The following experiments are carried out on these two real data sets to analyze and verify the above theories and algorithms.

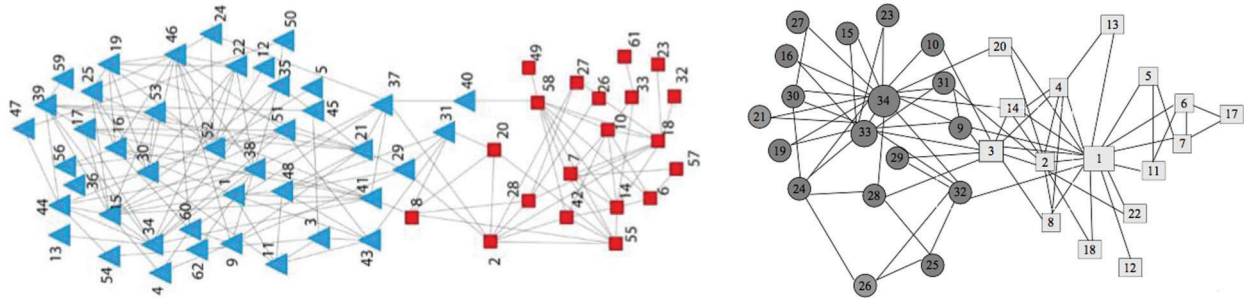


Figure 6: Dolphin social network and karate club network

4.2 Experimental Process

4.2.1 Unit Subgroup Generation

According to algorithm 1, based on the degree of nodes in the graph network, from the high to low order, the unit subgroup with the node as the core is generated successively until all nodes in the graph network are contained in the unit subgroups. Dolphin social network generates 43 unit subgroups, and karate club network generates 13 subgroups. As shown in Fig. 7.

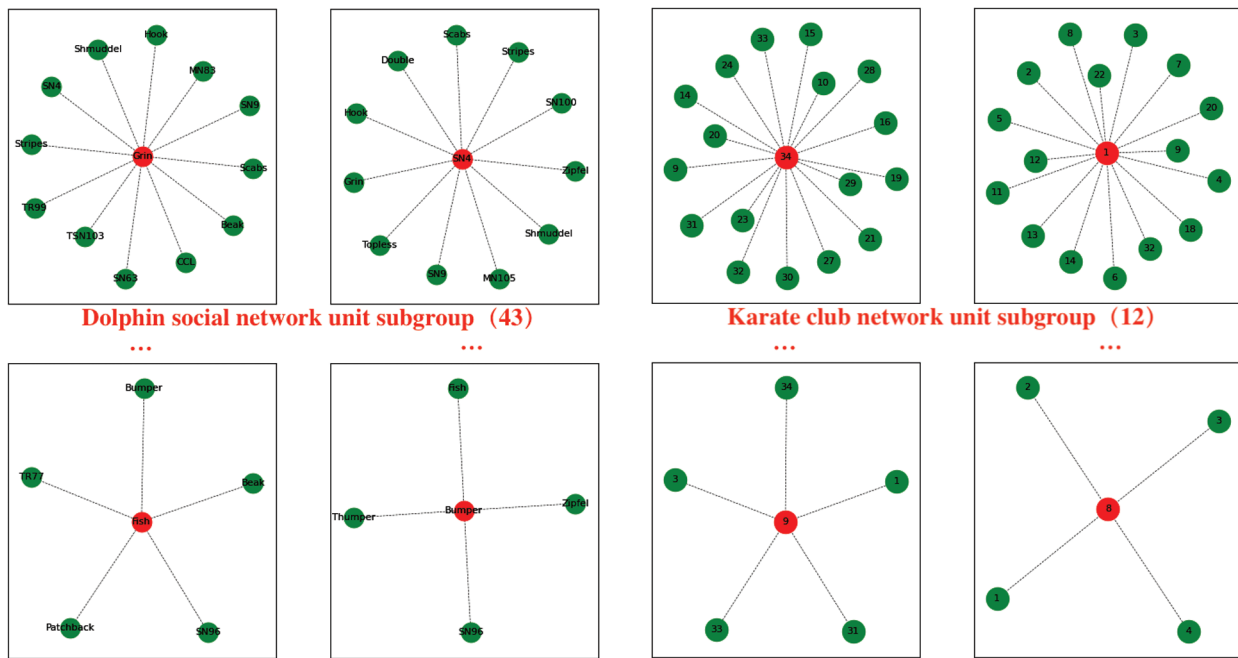


Figure 7: Unit subgroups of dolphin social network and karate club network

These unit subgroups are generated according to the clustering centrality (node degree), covering all nodes in the graph network. They are the basic elements of graph network subgroups.

4.2.2 Graph Network Subgroup Extension

According to algorithm 1, based on unit subgroups, the unit subgroups with high aggregation degree are selected in the order of node degree from high to low, and the k-order expansion is carried out until it can no longer be extended. After expansion, the dolphin network is expanded from 43 unit subgroups to

8 subgroups, of which 6 are 2-order subgroups and 2 unit subgroups are not expanded, indicating that the two unit subgroups tend to be separate groups, as shown in Fig. 8; The Karate club network is expanded from 12 unit subgroups to two 2-order subgroups. The expanded results are shown in Fig. 9.

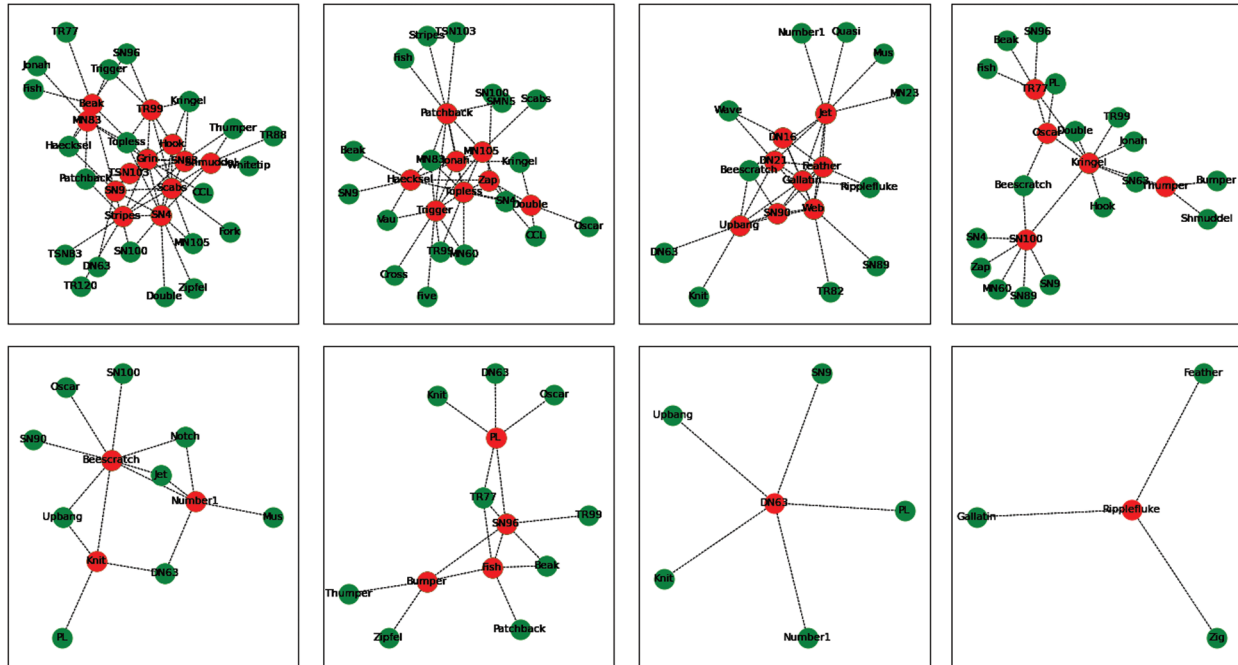


Figure 8: Eight subgroups of dolphin social network(after expansion)

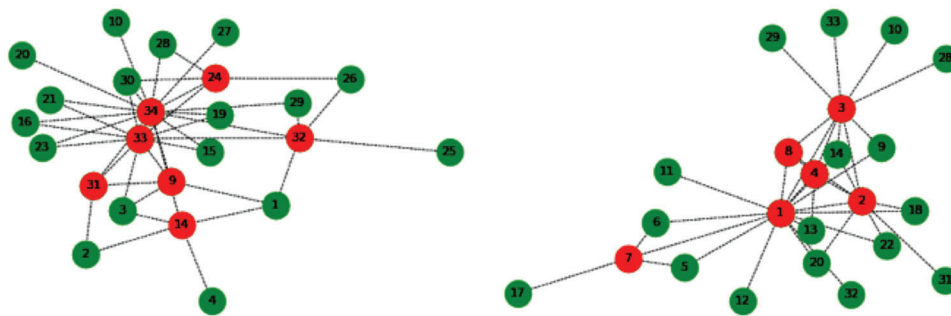


Figure 9: Two subgroups of the karate club network(after expansion)

4.2.3 Synthesis and Optimization of K-order Subgroups

If the generated k-order subgroups have more common nodes, further synthesis and optimization of k-order subgroups can be considered. The synthesized and optimized subgroups should have less connectivity, that is, there are fewer common nodes among subgroups. For Karate clubs, since the generated 2-order subgroups have reached the minimum number of graph network divisions, they will not be merged, and only the nodes between subgroups will be adjusted and optimized. The adjustment and optimization algorithm and idea are shown in the previous section. Subgroup structure recognition results are shown in Fig. 10.

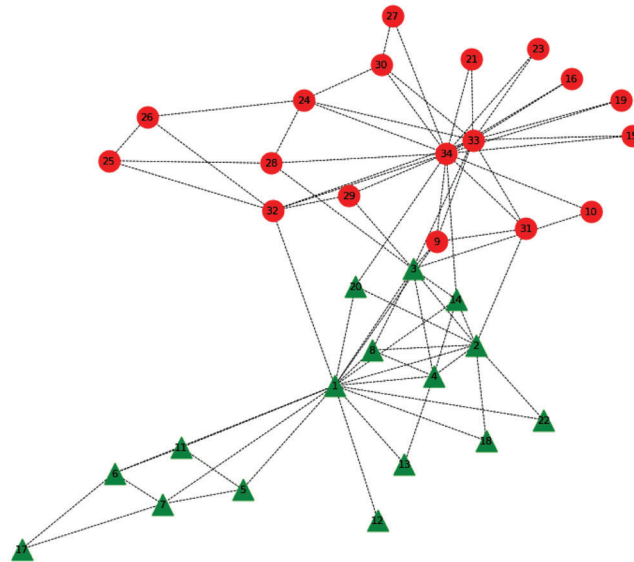


Figure 10: Two subgroup structures of the karate club network

For dolphin social network, eight subgroups are further merged into four subgroups by aggregation coefficient $K(K=0.6, K$ is a hyperparameter, it can be determined by observing the common nodes among subgroups in advance), two of which are unexpanded unit subgroups. According to algorithm 2, the number of common nodes of two unit subgroups and composite subgroups are compared respectively, and the unit subgroup is merged with the subgroup with more common nodes. After optimization and adjustment, eight subgroups are combined into two subgroups. The results are shown in Fig. 11. The coincident nodes among subgroups are processed according to the principle of coincident node processing. Subgroup structure recognition results are shown in Fig. 12.

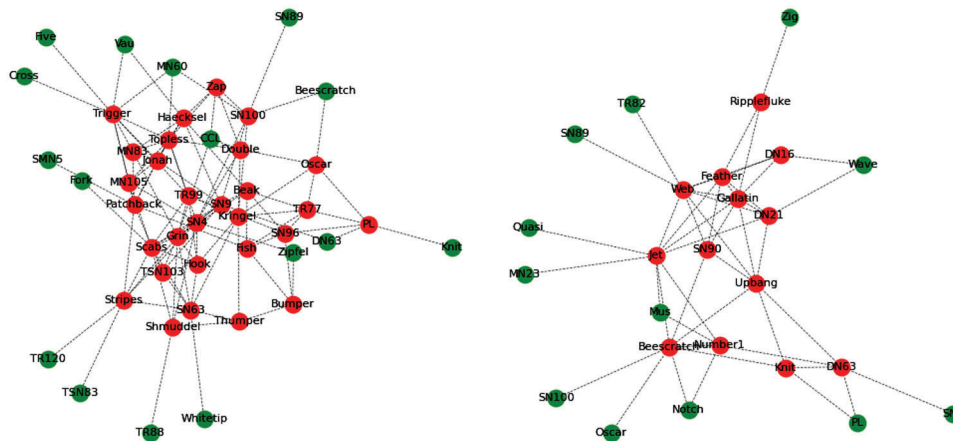


Figure 11: Two subgroups of the dolphin social network(after combined)

4.3 Analysis of Experimental Results

The experimental results show that the method based on unit subgroup expansion and core node signal propagation can effectively recognize the subgroup structure of the sensor graph network. The identification results are consistent with the actual subgroup structure of graph network. At the same time, the core node

unit structure and core nodes in graph network, and the universal significance of unit patulousness and the continuous propagation of the core node signals in the process of subgroup structure identification, which can be well interpreted. The research results can be used in the identification of sensor network structure and the core node transmission path discovery and so on. Compared with existing similar studies, the research process, methods and results of this paper are better interpretable and enlightening. In the future, further research and attempts will be made in the following aspects: dynamic adjustment and optimization of sensor network structure, message exchange model among sensor network subgroups, intelligent cooperation among sensor network subgroups and so on.

Funding Statement: This research is supported by the Natural Science Foundation Project of Fujian Provincial Department of Science and Technology (Grant No. 2020J01385), Digital Fujian Industrial Energy Big Data Research Institute (Grant No. KB180045), Provincial Key Laboratory of Industrial Big Data Analysis and Application (Grant No. KB180029).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] S. Kaur and V. K. Joshi, "Hybrid soft computing technique based trust evaluation protocol for wireless sensor networks," *Intelligent Automation & Soft Computing*, vol. 26, no. 2, pp. 217–226, 2020.
- [2] A. A. Hady, "Duty cycling centralized hierarchical routing protocol with content analysis duty cycling mechanism for wireless sensor networks," *Computer Systems Science and Engineering*, vol. 35, no. 5, pp. 347–355, 2020.
- [3] L. Kou, Y. Shi, L. Zhang, D. Liu and Q. Yang, "A lightweight three-factor user authentication protocol for the information perception of iot," *Computers, Materials & Continua*, vol. 58, no. 2, pp. 545–565, 2019.
- [4] C. T. Poomagal, G. A. Sathish Kumar and D. Mehta, "Multi level key exchange and encryption protocol for internet of things (iot)," *Computer Systems Science and Engineering*, vol. 35, no. 1, pp. 51–63, 2020.
- [5] J. Wang, W. Chen, L. Wang, Y. Ren and R. S. Sherratt, "Blockchain-based data storage mechanism for industrial internet of things," *Intelligent Automation & Soft Computing*, vol. 26, no. 5, pp. 1157–1172, 2020.
- [6] W. He, S. Guo, Y. Liang, R. Ma, X. Qiu *et al.*, "Qos-aware and resource-efficient dynamic slicing mechanism for internet of things," *Computers, Materials & Continua*, vol. 61, no. 3, pp. 1345–1364, 2019.
- [7] M. Okhovvat and M. R. Kangavari, "Tslbs: A time-sensitive and load balanced scheduling approach to wireless sensor actor networks," *Computer Systems Science and Engineering*, vol. 34, no. 1, pp. 13–21, 2019.
- [8] W. Zhang, W. Fang, Q. Zhao, X. Ji, "Energy efficiency in internet of things: An overview," *Computers, Materials & Continua*, vol. 63, no. 2, pp. 787–811, 2020.
- [9] D. Stavros I, M. Eirini and K. J. Derek, "The impact of graph construction scheme and community detection algorithm on the repeatability of community and hub identification in structural brain networks," *Human Brain Mapping*, vol. 42, no. 13, pp. 4261–4280, 2021.
- [10] J. S. Hossein, A. Qomi, A. Mahdi, A. Masoud and Y. Naser, "An information theoretic approach to link prediction in multiplex networks," *Scientific Reports*, vol. 11, no. 1, pp. 1–21, 2021.
- [11] J. Zhang, H. Liu and Z. Wen, "A sparse completely positive relaxation of the modulatory maximization for community detection," *SLAM Journal on Scientific Computing*, vol. 40, no. 5, pp. A3091–A3120, 2018.
- [12] K. He, Y. Li, S. Soundarajan and J. E. Hopcroft, "Hidden community detection in social networks," *Information Sciences: An International Journal*, vol. 425, pp. 92–106, 2018.
- [13] Y. T. Cui, Q. Niu and Z. X. Wang, "Semi-supervised spectral clustering approach for community detection based on signal transmission," *Computer Engineering & Design*, vol. 39, no. 5, pp. 9–13, 2018.
- [14] X. Q. Zhang, X. D. An and F. Y. Cao, "Detection community from bipartite network based on spectral clustering," *Computer Science*, vol. 46, no. 4, pp. 216–221, 2019.
- [15] W. P. Zhang, C. H. Che and Y. H. Qian, "A Two-stage community detection algorithm based on label propagation," *Computer Research and Development*, vol. 55, no. 9, pp. 1959–1971, 2018.

- [16] L. Li and L. Ni, "Community detection for label propagation with modularity optimization," *Computer Systems & Applications*, vol. 25, no. 9, pp. 212–215, 2016.
- [17] J. J. Whang, D. F. Gleich and I. S. Dhillon, "Overlapping community detection using neighborhood-inflated seed expansion," *IEEE Transaction on Knowledge and Data Engineering*, vol. 28, no. 5, pp. 1272–1284, 2016.
- [18] Y. Li, J. He, Y. X. Wu and R. J. Lv, "Overlapping community discovery method based on Two expansions of seeds," *Symmetry*, vol. 13, no. 1, pp. 1–19, 2020.
- [19] Y. X. Li, K. He, D. Bindel and J. E. Hopcroft, "Uncovering the small community structure in large networks: A local spectral approach," in *Proc. Proc. of the 24th Int. Conf. on World Wide Web*, Geneva, Switzerland, pp. 658–668, 2015.
- [20] F. Chung and O. Simpson, "Distributed algorithms for finding local clusters using heat kernel pagerank," in *Proc. Intertional Workshop on Algorithms and Models for the Web-Graph*, Springer, Cham, pp. 177–189, 2015.
- [21] I. M. Kloumann, J. Ugander and J. Kleingerg, "Block models and personalized PageRank," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 114, no. 1, pp. 33–38, 2017.
- [22] F. Chung and O. Simson, "Computing heat kernel pagerank and a local clusering algorithm," *European Journal of Combinatorics*, vol. 68, pp. 96–119, 2018.