doi:10.21311/001.39.10.40

Complex Network Anomaly Recognition Based on Multi-source Large Data

Yongxiang Lang
Department of Information Engineering, Chongqing City Management College, Chongqing 401331, China

Zheng Qin
College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, Hunan, China

Abstract
The balance factor of the number of societies and the smoothness of time is the biggest problem of the complex network anomaly recognition algorithm which integrates multiple sources of large data. Fuzzy network anomaly recognition algorithm based on recursive multi-source large data optimization is proposed, using the idea of multi-source large data genetic algorithm, the idea of multi-source large data is transformed into the optimization of multi-source large data genetic algorithm to ensure the quality of anomaly recognition at the current time, and the current anomaly recognition result can be consistent compared with network structure of the former time. Deman recursion is added to the algorithm in the initialization process to improve the quality of anomaly recognition of the initial individual. Mutation algorithm based on recursion is proposed to enhance the anomaly recognition effect and the convergence speed of the algorithm. At the same time, the combination of multi-source large data genetic algorithm with the recursive algorithm makes the algorithm highly extensible and the running time grows linearly with the increasing number of nodes or edges. By using the proposed algorithm to compare and experiment with the existing excellent algorithms in simulation datasets and real data sets. Experimental results show the proposed algorithm has both good anomaly recognition effects, but also good scalability.

Key words: Multi-source Large Data, Deman Recursion, Complex Network, Anomaly Recognition.

1. INTRODUCTION

In recent years, compared with the static network, the dynamic characteristics of social networks have been widely concerned, a large number of complex networks have been proposed, such as information exchange network and the scientists cooperation network (Lazer, Kennedy and King, 2014), social networks, etc. All complex systems have some dynamic characteristics. So it is a reasonable and effective way to simulate these complex networks into complex networks. Nowadays, complex network research methods are widely used in science and technology networks, biological networks and social networks to identify and describe the mutual relation among different things and individuals, such as the relationship between the biomolecules, cooperative relations, etc. and a community structure formed by a plurality of bodies such as the circle of friends, the biosphere. Figure 1 shows the network with three community structures, The value of the node connection represents the weight of the edge. For most networks, their topology will change significantly over time, such as each person's circle of friends will change over time and structural changes occur.

In the complex networks, community structure recognition is becoming an important research topic, and it has a wide range of applications in real society (Chen and Chiang, 2012; Boyd and Crawford, 2012), such as information impact analysis, customer recommendation, etc. The dynamic characteristics of complex networks mean as time changes, the structure of the community will also change accordingly. In the community structure, the association between the nodes within the community is relatively close, in the association between the nodes is relatively sparse. In the literature (Jagadish, Gehrke and Labrinidis, 2014; Margolis, Derr and Dunn, 2014), the community structure of the association is relatively small; the problem of community structure mining on complex networks is expressed as observing the connections between some individuals in the network from a discrete time axis, which is, observing the subgraphs at continuous time points. Structural changes can be understood as the majority of these sub-graph formed by the structure.

Incremental anomaly recognition (Davenport and Barth and Bean, 2012) and multi-source large data (Bhuyan, Bhattacharyya and Kalita, 2014) are two main methods to study complex network anomaly recognition. The basic idea of an incremental anomaly recognition algorithm only for abnormal recognition for the first time point of the network, for a network at a later point in time, according to the network slow variation characteristics, the anomaly recognition result at the previous point in time is taken as the basis, and the anomaly recognition result is adjusted locally by using the features of the current time point network. Finally, the result of anomaly
recognition is smooth, the represented algorithm are IA-MCS (Palmieri, Fiore and Castiglione, 2014), GraphScope (Flores, Calderon and Antolino, 2015), etc. The method of incremental anomaly recognition is to some extent sacrifice the quality of anomaly recognition to obtain smaller time complexity, while the multi-source large data method can also be used to consider the effects of anomaly recognition quality and time smoothness on the results, and to ensure the quality of anomaly recognition and make the recognition result closer to the real social structure. Lin et al (Muniyandi, Rajeswari and Rajaram, 2012) proposed FacetNet framework, which is the most classical multi-source large data algorithm, the algorithm uses KL-divergence algorithm to define the quality of the snapshot and the historical cost. We combine the anomaly recognition and the community evolution as a whole. The \( t \) time-history data and the historical community structure influence the \( t \) time-of-day community structure, so the community structure obtained in this algorithm is superior in noise immunity and rationality. In each of the iteration, the value of the approximate structure is updated to decrease historical cost, and finally converges to a local optimal solution.

**Figure 1.** The network which has three community structures

Kim et al. (Yasami and Mozaffari, 2010) have proposed an evolutionary approach to the fusion of particles and density, which initially constructs a complex network as a series of nano-communities defined as a subset of closely connected particles in a particle swarm. Then, the author uses the fusion density anomaly recognition method and the cost embedding technique to achieve the temporal smoothness of the community structure, and the algorithm does not depend on the specific anomaly recognition method and the definition of the similarity between the nodes, The author defines the similarity connection between adjacent nodes, and transforms the complex network into a complete multi-part graph, which is a complex graph corresponding to a time-point network. This algorithm solves the problems of the two problems of the most previous algorithms are as follows: (1) the number of communities is constant at each time (2) the time smoothing is achieved in an iterative manner and the efficiency is reduced.

However, in the above method, there are two common problems: (1) the number of associations to determine (2) used to achieve the balance of time smoothness factor determination. FacetNet (Muniyandi, Rajeswari and Rajaram, 2012) algorithm can only be used to identify the fixed number of communities of the complex network, that is, throughout the network time series, the number of associations will not change. Although the FacetNet extended version to make up for some of the previous problems, for example, can increase the node and delete the node, you can also deal with adjacent moments The number of changes, but the algorithm itself also has shortcomings, for example, requires many iterations to make the matrix to converge. Therefore, FacetNet extended version is not suitable for large-scale data processing. And Kim's algorithm is fusion density anomaly recognition algorithm, so you need to set the density parameter in advance.

Another problem is that these algorithms need to set the balance factor in advance to achieve the time smoothness. Folino et al. (Tartakovsky and Polunchenko, 2013) proposed a multi-source large data algorithm (DYNMOGA) for the complex network anomaly recognition problem. The framework of anomaly recognition is a multi-source large-data genetic algorithm, which effectively balances the time cost and the historical cost, and improves the quality of anomaly recognition significantly, and it can automatically identify the number of communities without balance factor. Through a large number of experiments, Algorithm is superior to classical FacetNet algorithm in the dynamic network community discovery based on evolutionary clustering, the time complexity and clustering precision are better than those of classic FacetNet algorithm. But because
DYNMOGA algorithm adopts the coding method of genetic algorithm and fusion graph, the time complexity is high, which cannot handle the large-scale network well. Finally, in the multi-source large data optimization method which is considered to take account of the dynamic characteristics of the network, the randomness of the individual is larger, increase the large number of population and the number of iterations can be used to obtain the quality of the anomaly, and the quality is sacrificed at the same time. DYNMOGA algorithm adopts the encoding method of fusing the adjacent position, that is, each node uses its neighbor node number to represent the gene value, the node and the gene value indicates that there are edges between two nodes. Decoding process is the division of the corresponding community and determine the number of associations. Adjacent location encoding fusion in the decoding process takes time $O(n \log n)$. However, the time complexity of the entire DYNMOGA algorithm $O\left( (gp \log p) \times (n \log n + m) \right)$ can be seen, the decoding time in the entire algorithm accounted for a large part of the impact of the efficiency of the algorithm.

Therefore, in order to solve the problem of high randomness and time complexity in DYNMOGA algorithm, this paper introduces the idea of Deemann recursion and combines the dynamic characteristics of the network, that is, considering the time cost and the historical cost, and apply it to in the complex network anomaly recognition, a multi-source and large-data algorithm LDMGA with multi-source and large-data optimization is proposed.

The main contributions of this paper include the following three aspects:

1. In the process of initializing the individual, the Derman recursive algorithm of the fusion node degree is introduced, which makes the initial community have certain precision and enhances the quality of anomaly recognition.

2. The fusion algorithm of fusion DeMart recursive algorithm is proposed to further enhance the quality of anomaly recognition and accelerate the convergence rate of the algorithm.

3. In the multi-source large data genetic algorithm combined with the Derman recursive algorithm, the algorithm can be extended, and the running time of the algorithm increases linearly with the number of nodes or edges.

The first section introduces the multi-source and large-data multi-source data model. Section 2 gives the introduction of LDMGA, which is an anomaly recognition algorithm of multi-source large-data optimization based on the Derman recursive algorithm. Section 3 mainly introduces the LDMGA algorithm, the whole process is introduced and time complexity analysis; Section 4 of this algorithm for experimental testing and analysis; Section 5 concludes the full text.

Figure 2. Basic framework of LDMGA algorithm
2. ANOMALY RECOGNITION ALGORITHM BASED ON DERMAN RECURSIVE ALGORITHM FOR MULTI-SOURCE LARGE DATA OPTIMIZATION LDMGA

2.1. Integration of Multi-Source Large Data Optimization Anomaly Recognition Algorithm Framework

The basic framework of LDMGA algorithm is genetic algorithm. Genetic algorithm (Celenk, Conley and Willis, 2010) is a kind of adaptable search method. In genetic algorithm, the solution set of multi-source large data optimization problem is defined as the best individual in the population, Represents a possible solution. The number of individuals in the population represents the size of the population. Each individual is a collection of multiple genes, can be understood as a combination of permutations of genes, different combinations determine the individual's community structure. Therefore, the algorithm in the initial stage, the individual needs to transform the individual realization from the phenotype to the genotype, i.e., the coding, and the individual evolves continuously in successive offspring. Often, the offspring are generally produced by crossing the two parental individuals, Gene structure, and then mutation operation of the parent individual gene mutation to generate a better structure in each generation, the individual should be the fitness value of the calculation, the objective function of the calculation, a higher degree of fitness Will be selected into the next generation of iterations. With several generations, the individual will tend to meet a given condition, i.e., a good community structure, and the final individual is considered to be optimal or next to optimal for all objective functions. The basic framework of LDMGA algorithm is shown in Figure 2.

LDMGA algorithm process diagram is shown in Figure 3, the value of the node connection denotes the edge weight. Figure 3 (a) shows the initial input network. In the initialization process, the network is divided into different community structure, generate a different community structure as shown in Fig.3 (c), then the initial population is selected, crossover and mutation to select the optimal community structure, as shown in Fig.3 (c).

![Figure 3. LDMGA algorithm process diagram](image)

2.2. Individual Encoding and Decoding

At present, in the community structure recognition algorithm, there are mainly string encoding (Monroe, Pan and Roberts, 2015; Banu, Bhandarkar and Nambiar, 2013) and fusion map coding method. Compared with the encoding method of the figure, the string encoding method is more intuitive in representing the community structure and efficient, so this paper uses a string encoding. An arbitrary partition of a network is called an individual, contains n gene \( g_1, g_2, \ldots, g_n \) n is the number of nodes. Each gene corresponds to a value \( j \). These genes constitute the network, and each value \( J \) corresponds to the first \( i \) gene \( g_1 \), and \( j \) represents the label of the community to which the gene belongs, which means that the genes with the same tag belong to the same community. Figure 4 shows a network partition and the corresponding coding.

In the string representation, the network node belongs to only one identifier, that is to say, the nodes with the same label in different individuals may not belong to the same community. As shown in Figure 4, the network may have two individuals \( \{1,1,1,2,2,2,2\} \) \( \{2,2,2,2,3,3,3\} \). There are two societies \( \{1,2,3\} \) \( \{4,5,6,7\} \) in the first individual, and the second individual there are two societies \( \{1,2,3,4\} \) \( \{5,6,7\} \) and if the two individuals conduct the cross-operation, then the cross-result may be \( \{2,2,2,2,2,2\} \), then the entire community structure is destroyed. So, this decoding process for any individual, its initial label \( L(i) (i = 1, \ldots, n) \) assuming \( L(1) = 1 \)
if \( L(2) = L(1) \) then \( L(2) = 1 \) if \( L(2) \neq L(1) \) then \( L(2) = 2 \) and so on \( S(k) \neq S(j), (j = 1, 2, \ldots, k - 1) \) at this time \( p - 1 \) it is the current maximum community label value, then \( S(k) = p \) the individual \((1,1,1,2,2,2,3,3,3)\) is decoded in the whole decoding process, and then \((1,1,1,2,2,2)\), and then \((1,1,1,2,2)\). And then conduct cross-operation, easy to retain the community structure in decoding, the node label represent the number of the node belongs to the community.

![Figure 4](image)

**Figure 4.** 7-Node network is divided into two communities \{1, 2, 3\} and \{4, 5, 6, 7\} and the corresponding code denotation

2.3. Single Crossover Strategy

As the algorithm decoding process to some extent to resolve the incompatibility between the different labels of the situation. In order to further preserve the good community structure, this paper adopts the single crossover strategy, and the crossover strategy is shown in Table 1, where S is the number of the node. In Table 1, there is a network with 6 nodes. Suppose node 2 is selected at random. In node A, the node label of node 2 \( \{1,2,6\} \) will be propagated to node B \( \{1,2,6\} \), that is node A in node 2 is updated from the label 1, so as to obtain of the new individual C after the cross. This cross-operation can be A in the community structure of information transmitted to B. In this paper, the implementation of each cross-operation is done twice, once A spread to the B, once B to A.

<table>
<thead>
<tr>
<th>S(node)</th>
<th>A(source)</th>
<th>B(target)</th>
<th>C(new)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>→ 1</td>
<td>→ 1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>→ 2</td>
<td>→ 1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>→ 1</td>
<td>→ 1</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>→ 2</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>→ 2</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>→ 1</td>
<td>→ 1</td>
</tr>
</tbody>
</table>

2.4. LDMGA Algorithm Objective Function

Objective function: As described in (1) above, the algorithm focuses on optimizing the quality of the snapshot \( SC \) and optimizing the history overhead \( TC \), so as to finally optimize \( cost \) because the snapshot quality \( SC \) measures the quality of the community structure at the moment \( t \), so an objective function is needed to maximize the number of edges in each community, the number of edges between communities is minimized. Therefore, this paper adopts the standard-modularity \( Q \) widely used in the field of community structure identification.

Network \( G_t = (V_t, E_t) \) In time \( t \) there are \( n \) nodes and \( m \) edges, Its social structure is recorded as: \( C = \{C_1, C_2, \ldots, C_k\} \), \( k \) For the number of associations. \( l_s \) denotes the community \( C_s \). The number of edges between all the nodes, \( d_s \). Represents the sum of the node degrees in the community.

The modularity \( Q \) is defined as follows:

\[
Q = \sum_{s=1}^{k} \left[ \frac{l_s - (d_s/2m)^2}{m} \right]
\]  

(1)
In the formula of modularity \( Q \), the first part represents the probability of edges in a community, and the second part shows the probabilities of the edges in the whole network if the edges are randomly assigned without considering the community structure.

The second objective function must minimize the historical overhead \( TC \). This paper uses standardized mutual information Normalized Mutual Information (NMI), Using a matrix to measure the time of the community structure and the previous moment of the community structure of the similarity of the assumption that a network of two partitions \( A = \{ A_1, A_2, \ldots, A_p \} \) and \( B = \{ B_1, B_2, \ldots, B_q \} \), \( C \) is a matrix, the elements \( C_{ij} \) is at the same time in the community \( A_i \in A \) and the number of the nodes in the community \( B_j \in B \).

The definition of NMI is:

\[
NMI(A,B) = \frac{-2 \sum_{i=1}^{C_p} \sum_{j=1}^{C_q} C_{ij} \log(C_{ij}/C_p C_q)}{\sum_{i=1}^{C_p} C_i \log(C_i/N) + \sum_{j=1}^{C_q} C_j \log(C_j/N)}
\] (2)

In this formula, \( C_p \) represents the number of organizations in the division \( A \), \( C_q \) said the number of organizations in the division \( B \), \( C_i \) said the sum of the matrix \( C \), \( C_j \) said the sum of the columns in the matrix, \( N \) is the number of nodes. If \( A = B \) then \( NMI(A,B) = 1 \), if \( A \) and \( B \) completely different \( NMI(A,B) = 0 \). So the second goal is to maximize \( NMI(C_{t+1}) \) at the time \( t \).

3. EXPERIMENTAL RESULTS AND ANALYSIS

In the algorithm of multi-source large data, the most classic is the FacetNet framework proposed by Lin et al. (Muniyandi and Rajeswari and Rajaram, 2012) and the multi-source large data method of fusion particle and density proposed by Kim et al.11 Folino et al. The DYNMOGA algorithm is better than the previous two algorithms by comparing the two algorithms to identify the effects of both anomaly recognition and time smoothing. In this section, the proposed LDMGA algorithm and DYNMOGA algorithm, FacetNet algorithm and algorithm proposed by Kim et al. other algorithms for comprehensive evaluation.

3.1. Experimental Setup

LDMGA algorithm, DYNMOGA algorithm and FacetNet algorithm are implemented using Matlab, which, DYNMOGA algorithm and FacetNet algorithm source code provided by the author. The hardware environment of this experiment: CPU 2.3 GHz Intel (R) Core (TM) i3-2350M, memory 4G; Software environment for Windows 7: Matlab R2008b.

3.1.1. Parameter setting

LDMGA algorithm uses genetic algorithm, so the choice of parameters is very important in evolutionary algorithm. This paper uses trial-and-error method to experiment by changing crossover and mutation probability. The data set network generated in this paper consists of 128 nodes, and each node has a fixed average degree \( \text{avgDegree} = 16 \), and each node \( z = 5 \) is connected with nodes outside the community.

![Figure 5. Standardized mutual information values for different crossover probabilities and mutation probabilities](image-url)
As can be seen from Figure 6, the standardized mutual information NMI value does not show significant change. In view of the general use of high crossover probability, low probability of mutation, so set the crossover probability is 0.8, the mutation probability is 0.2. The number of population is 100, the iteration number is 100, and the result is the value of running once. DYNMOGA (Tartakovsky and Polunchenko, 2013) the parameters of the experiment are as follows: the crossover probability is 0.8, the mutation probability is 0.2, the number of population is 100, the number of iterations is 100, and the result is 50 running times and gets averages. FacetNet algorithm \( \alpha = 0.8 \) is set up.

3.1.2. Evaluation Indicators

In this paper, two authentication functions are used to evaluate the quality of the results, NMI and Error Rate, and the NMI function has been introduced earlier. In Error Rate calculation, we first establish an identity matrix \( Z \), \( Z \) for the matrix \( n \times k \), \( n \) the number of nodes, \( k \) is the community number And a similar sign moment \( G \), said the real community. Error Rate is defined as \( \| ZZT − GGT \| \) that community \( Z \) and the real community \( G \) in the community structure on the distance.

3.2. Dataset # 1

Dataset # 1 considers some important events to mark the evolution of complex networks. The method of generating datasets in this paper generates four data sets for a total of 20 time points. The parameters are set to 1000 nodes, the average degree of each node is 15, 50, the mixing parameter 0.2, i.e., the probability of edges between societies.

Data Merging and splitting: At each point in time, 10% of the communities are split, 10% of the communities are selected, and merged.

Data Expansion and contraction: 10% of the community randomly selected for expansion or contraction of the community size, the proportion is 25%. When the expansion happened, the new nodes were randomly selected from other societies.

Intermittent communities: 10% of societies are hidden from the first point in time.

Data Birth and death: From the second point in time, 10% of new societies were created and 10% of existing societies were randomly removed by moving nodes from existing communities.

![Figure 6](image-url)

**Figure 6.** Merging and splitting (a) Normalized Mutual Information (b) Error Rate

From the experimental results in Figure 6 (a) and (b), the LDMGA algorithm has a better NMI value and Error value than the DYNMOGA algorithm and the FacetNet algorithm. In the process of merging and merging of the community, the tagged LDMGA algorithm can accurately identify the community structure, so the value of NMI has been close to 1, The NMI value of DYNMOGA algorithm is always around 0.95. From Fig.6 (a), it can be seen that the NMI value of FacetNet algorithm is always decreasing, and the decreasing range of NMI is always decreasing, the error rate of DYNMOGA algorithm is about 7000. The error rate of FacetNet algorithm is on the rise (Fig.6 (b)), and the error rate of DYNMOGA algorithm is about 7000. The error rate of FacetNet algorithm is increasing), The FacetNet algorithm is not suitable for the anomaly recognition of the complex network. It is known from the experimental data that DYNMOGA algorithm, there are more nodes in the wrong community, or there is a small community is not merged.
It can be seen from Figure 7 (a) (b) that the results of the three algorithms are similar to those of the data set Merging and splitting, except that the difference between the FacetNet algorithm and the DYNMOGA algorithm is narrowed. When the community is expanding or contracting Figure 7 (a), LDMGA algorithm to identify the community than the DYNMOGA algorithm and FacetNet algorithm community structure is closer to the real community structure from the NMI results have been 1 or close to 1, the error rate close to 0 or 0 from this it can be seen, LDMGA algorithm is very suitable for this type of network. While the average error rate of DYNMOGA algorithm in 5000 or so, NMI value of 0.97 or so, indicating that there are some nodes in the wrong community, while FacetNet algorithm in this complex network performance Down, when the time point 17, the performance decreased significantly.

It can be seen from Figure 8 (a) (b) that the NMI values of the LDMGA algorithm are 1 and the Error value is 0. At 20 time points, this means that at each time points exist where the community is hidden. LDMGA algorithm is still able to identify the community structure of this type of network. Although the NMI value of DYNMOGA algorithm is close to 1 (Figure 9 (a)), but the LDMGA algorithm identification (Fig. 8 (b)), which is significantly lower than that of the LDMGA.
algorithm. In Fig. 8 (b), the NMI value of FacetNet algorithm is about 0.95, and the error rate is higher, significantly lower than the performance of LDMGA algorithm.

It can be seen from Figure 9 (a) and (b) that although the results of DYNMOGA algorithm are similar to those of LDMGA algorithm, the running results of LDMGA algorithm are better than that of DYNMOGA algorithm. In the networks of new societies and old societies, (Fig. 9 (a)), it can be seen that the community structure identified by the LDMGA algorithm is closer to the real community structure than the DYNMOGA algorithm. Especially from the Error value, it can be seen that the DYNMOGA error rate is about 3000 and the LDMGA algorithm error rate is close to 0. This indicates that the LDMGA algorithm is very suitable for this type of network. While FacetNet algorithm performance with the community generation and disappearance, NMI value in the continuous decline, Error value is increasing to be big.

3.3. Power-law Network

The LER network data used in this paper, The network has 1000 nodes with an average node degree of 20, a maximum node degree of 50, a power exponent of degree distribution of -2, a community size distribution of -1, and a mixing parameter of 0.3. The network is having only one node of the maximum degree of 50. 70% its node degrees are below the average node degree of 20.

In order to introduce a complex network, five time points are selected, 10% of the communities are selected at random, and the splitting (time point 2, 4) and merging (time points 3, 5) are repeated (see Figure 10 (a)). LDMGA and the algorithm DYNMOGA, FacetNet algorithm NMI and Error results. Experimental results show that from the first point in time, LDMGA algorithm is better than DYNMOGA algorithm and FacetNet algorithm results, LDMGA algorithm and FacetNet algorithm NMI results the same trend, but LDMGA Algorithm can identify the community structure well, and the value of NMI has been not less than 0.98. Although DYNMOGA algorithm from the second point in time, NMI results stable at 0.96, but LDMGA algorithm error rate (Figure (b)) was significantly lower. The DYNMOGA algorithm and FacetNet algorithm prove that the LDMGA algorithm can accurately identify the community to which the node belongs.

![Figure 10. Power-law network (a) Normalized mutual information (b) Error rate](image)

3.4. Scalability Analysis

The major obstacle to solving the optimization problem with genetic algorithms is the high computational time, and one of the main problems of evolutionary algorithms is the repetition of the fitness function, which can become serious when the population is large is a multi-source large data optimization method. From the previous time complexity analysis can be seen in the algorithm in this paper, the fitness calculation is very efficient for large networks are also very effective.

<table>
<thead>
<tr>
<th>Table 2. Algorithm running time p=50</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDMGA/ DYNMOGA</td>
</tr>
<tr>
<td>p=50</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
data. Furthermore, they are superior to the current excellent algorithm. Most be worth mentioning is that, results show that the algorithm proposed in this algorithm of the fusion tag enhances the community structure and reduces the running time. Experimental source large data optimization can be applied in every time point, which provides a good balance between good community structure on complex networks and satisfy the requirement of time points. Adding the tag algorithm to the multi-source large data optimization algorithm is proposed, which can identify good community structure and further improve the running time of LDMGA algorithm. LDMGA algorithm time complexity is \(O(gp \log p \times (n+m))\), when the population number and iteration number, the running time of LDMGA algorithm increases linearly, so the algorithm is more extensible.

4. CONCLUSION

In this paper, a multi-source large data optimization algorithm is proposed, which can identify good community structure on complex networks and satisfy the requirement of time smoothness. The idea of multi-source large data optimization can be applied in every time point, which provides a good balance between good network structures based on the current point in time and small differences in network structure at continuous time points. Adding the tag algorithm to the multi-source large data optimization algorithm, the mutation algorithm of the fusion tag enhances the community structure and reduces the running time. Experimental results show that the algorithm proposed in this paper has good performance in both simulation data and real data. Furthermore, they are superior to the current excellent algorithm. Most be worth mentioning is that, LDMGA algorithm run-time obvious advantages DYNMOGA algorithm, more suitable for large-scale data mining.

REFERENCES


Table 3. Algorithm running time p=100

<table>
<thead>
<tr>
<th>LDMGA</th>
<th>DYNMOGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node 2^7</td>
<td>Node 2^8</td>
</tr>
<tr>
<td>p=50</td>
<td>p=100</td>
</tr>
<tr>
<td>4.816s/</td>
<td>10.673s/</td>
</tr>
<tr>
<td>5.8756s/</td>
<td>20.646s/</td>
</tr>
<tr>
<td>8.973s/</td>
<td>36.199s/</td>
</tr>
<tr>
<td>36.644s/</td>
<td>73.273s/</td>
</tr>
<tr>
<td>82.237s/</td>
<td>165.51s/</td>
</tr>
<tr>
<td>223.869s/</td>
<td>419.365s/</td>
</tr>
<tr>
<td>481.79s/</td>
<td>931.60s/</td>
</tr>
<tr>
<td>1201.82s/</td>
<td>2274.07s/</td>
</tr>
</tbody>
</table>

The network is divided into four communities, the average node degree of each point in time \(avgDegree = 16, z = 5\), 10% of the nodes are selected to enter other communities, and the number of nodes \(n\) is the range of change \{128, 256, 512, 1024, 2048, 4096, 8192, 16384\} , the corresponding range of \{1938, 4018, 8184, 16158, 33026, 65256, 131388, 285782\} the range of changes in the number of groups \(p\) is \{50, 100, 200\}, The range of iterations \(g\) is \{50, 100\} . Table 2 and 3 show the results is at a time point of the LDMGA algorithm and the running time DYNMOGA algorithm for different combinations \(p, g\). As can be seen from Tables 2 and 3, especially when the number of nodes doubled, the running time of LDMGA algorithm is significantly lower than DYNMOGA algorithm, DYNMOGA algorithm itself running time is more than 6 times the running time of LDMGA algorithm. LDMGA algorithm time complexity is \(O(gp \log p \times (n+m))\), when the population number and iteration number, the running time of LDMGA algorithm increases linearly, so the algorithm is more extensible.


