A Complex Network Clustering Algorithm based on Gene Expression Programming and Information Theory

Rong Tang\textsuperscript{1,2}, Changjie Tang, Jie Zuo, Ning Yang\textsuperscript{1}
\textsuperscript{1}School of Computer Science
Sichuan University
Chengdu, China
e-mail: tangrongcs@126.com, cjtang@scu.edu.cn

Kaikuo Xu\textsuperscript{2}
\textsuperscript{2}Department of Computer Science & Technology
Chengdu University of Information Technology
Chengdu, China
e-mail: kaikuoxu@gmail.com

Abstract—The problem of complex network clustering has been long studied and a satisfactory solution is still lack so far. By cohesion superiority of Gene Expression Programming and the information-theoretic topology compression framework, a novel algorithm, CNC-GEP, is proposed for complex network clustering. With no prior knowledge of the number of modules in networks, CNC-GEP not only determines the optimal number of modules but also resolves how to cluster nodes into that number of modules so that the best module structure is revealed for networks. CNC-GEP is empirically evaluated by running it on two benchmark networks and the empirical results show that our algorithm CNC-GEP can correctly resolve module structure for complex networks in certain successful rate and the average accuracy is even up to 99.35% for Zachary’s karate club network.

Keywords—complex network clustering; information theory; Gene Expression Programming

I. INTRODUCTION

Lots of complex systems in real world such as social network and collaboration network of scientists can be modeled as complex networks with nodes representing entities and edges representing relations. Complex networks can be considered at high level as a collection of communities consisting of subset of nodes within which the density of edges is higher than between them\cite{1}. As complex networks become larger and larger in scale, to understand their structure becomes a key issue to recognize complex networks. However, finding an optimal clustering solution for complex networks is generally believed to be NP-complete\cite{2}. Although various network clustering methods have been invented in recent years, a satisfactory solution is still lack due to the following challenges: (1) how to determine the optimal number of modules without prior knowledge of communities composing a complex network; (2) how to properly assign the nodes into the determined number of modules; (3) What is the proper criterion to quantify the strength of community structures for complex networks.

To solve clustering problems of such kind, Gene Expression Programming (GEP), a new member of genetic computation invented by Ferreira\cite{3}, is an approach that gives acceptably good solutions in many cases. Ferreira had successfully applied it on solving combinatorial optimization problems\cite{4}. Rosvall and Bergstrom proposed an information-theoretic framework to resolve community structures for complex networks\cite{5}. With no \textit{a priori}, the framework determines the number of modules in networks by following the principle of minimum description length (MDL)\cite{6, 7}, and finds out the best assignment of nodes by maximizing the mutual information between a network and its module descriptions.

By cohesion superiority of GEP and the information-theoretic framework, we propose a complex network clustering algorithm (CNC-GEP). With no prior knowledge of the number of modules in networks, CNC-GEP not only determines the optimal number of modules but also resolves how nodes are distributed into that number of modules so that the best module structure is revealed. In summary, the contributions of this paper are as follows:

- A new chromosomal organization is designed to model the module structure for complex networks in which a CN-gene represents a module and a multi-gene CN-chromosome encodes a module structure.
- Genetic modifications specific to complex network clustering are designed not only to maintain chromosomes always valid, but also to fulfill the operations of nodes’ moving and exchanging between modules.
- MDL and modularity\cite{2} is utilized as the fitness function in the evolution of CN-chromosome individuals. As a comparison, is also employed as a fitness function. CNC-GEP adopting either fitness function is run on benchmark networks so that their partition results are compared in terms of the accuracy and efficiency.

II. RELATED WORKS

Based on the betweenness centrality in 2003, Grivan and Newman proposed an algorithm GN to extract the natural community structure from complex networks\cite{2}. Modularity is firstly proposed to quantify the strength of community structure in\cite{2}, and later on it has been widely used to verify algorithms in\cite{5, 8, 9}. The modularity approach, however, needs the priori knowledge of the number of modules in networks. To find out the optimal number of modules
composing a network with no prior knowledge, Rosvall and Bergstrom proposed an information-theoretic compression framework [5].

Based on optimizing modularity, a genetic algorithm is proposed to detect communities with no priori knowledge of community structures of complex networks in [9]. However, our testing on Zachary’s karate club network shows that only maximizing the modularity cannot reveal its classic 2-module structure.

III. PRELIMINARIES

As in basic GEP[3], a gene is a linear strings consisting of terminals and functions, and a chromosome acting as the genotype consists of a set of genes. Chromosomes are subject to genetic modification in evolution, and are decoded into expression trees functioning as phenotype when subjected to genetic modification in evolution, and are genotype consists of a set of genes. Chromosomes are genetic modification operations, and the fitness functions.

A. Module Description

A complex network is modeled as an unweighted and undirected network X with n nodes and l links. A community structure of X into m modules, also called as the module description Y of X, is represented as in formula (1) [5].

\[ Y = \{a = \{a_1, a_2, \ldots, a_m\}, MM = \{l_{i1} \leq l_{in}, \text{M}, \text{O}, \text{M}, l_{m1} \leq l_{mn}\}\} \quad (1) \]

where a is the module assignment vector, and \( a_i \) is the \( i \)th module (\( l \leq i \leq m \)). MM, the module matrix, describes how the \( m \) modules are connected internally and externally, where \( l_{ij} \) is the number of links between nodes in module i and \( l_{jk} \) is the number of links that connects two nodes respectively located in module i and j (\( i \neq j \))[5].

B. Definitions

In our CNC-GEP, the indices of \( n \) nodes are presented as terminals. A modulization function \( M \) is designed to represent the grouping of nodes and \( M \) is the only function in CNC-GEP. Thus, we have terminal set \( T = \{0, 1, \ldots, n-1\} \) and function set \( F = \{M\} \). A CN-gene represents a module. It consists of a head — the modulization function \( M \), and a body — the subset of terminals. The length of a CN-gene is the number of the terminals in its body. A CN-gene is empty if its length is zero. A multi-gene CN-chromosome encodes a module structure for complex networks.

Definition 1. (CN-gene): A CN-gene is a 3-tuple, \( \{L, T, F\} \), where

- \( L \) (List) is a list with unfixed length;
- \( T \) (Terminal) is an alphabet, a subset of node terminals;
- \( F \) (Function) is a set of computing functions, \( F = \{M\} \).

Definition 2. (CN-Chromosome). Let \( m \) be the number of modules and \( n \) be the number of nodes in a complex network. A CN-Chromosome is a set of \( m \) CN-genes, \( C = \{g_1, g_2, \ldots, g_m\} \). A CN-chromosome is viable if each of its member CN-genes is non-empty; otherwise, a CN-Chromosome that contains any empty member CN-gene is nonviable.

Example 1: A complex network \( X_i \) consists of 9 nodes and 11 links. The best clustering for \( X_i \) is shown in Figure 1 where \( m = 3 \). \( X_i \) is represented by a CN-chromosome: \( M035M2467M18 \) and is expressed by a module description \( Y_i \) as in (2).

\[ Y_i = \{a = \{0, 3, 5\}, \{2, 4, 6, 7\}, \{1, 8\}\}, M = \begin{bmatrix} 3 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 1 \end{bmatrix} \] (2)

IV. METHODOLOGY

In this section, the decoding of CN-chromosomes, CNC-specific genetic operators, and the fitness functions, and the details of CNC-GEP are introduced.

Procedure: DecodeChromosome(C)

Input: A CN-chromosome C

Output: A modular description Y

1. Scan C to generate a set of CN-genes: \( \{g_0, g_1, \ldots, g_m\} \);
2. for each CN-gene \( g_i \) { 
3. if (\( g_i \) is an empty gene)
4. \( C \) is nonviable and be discarded;}
5. else
6. Group all its node terminals in module \( a_i \);}
7. \}
8. Obtain the parameters of module description
9. for each module \( a_i \) { 
10. Count its total degree \( d_i \);
11. Count its inlinks \( l_{i0} \);
12. Count outlinks between \( a_i \) and \( a_j \);
13. \( j \leftarrow j+1 \);
14. for each neighbor module \( a_i (j > i) \) of \( a_i \) { 
15. Count outlinks \( l_{ij} \) between \( a_i \) and \( a_j \);
16. \( l_{j0} \leftarrow l_{j0} + l_{ij} \);
17. } 
18. update the module matrix MM;
19. }
20. Output the modular description \( Y = \{a, MM\} \);

A. Decoding

The steps to decode a CN-chromosome into a module description are as follows:

- Scan a CN-chromosome sequentially from left to right to obtain the set of member CN-genes. The
CN-chromosome is nonviable and will be discarded if there exists any empty CN-gene.

- Generate the set of modules by decoding each CN-gene into a module.
- Generate the module description \( Y \). For every module \( a_i (0 \leq i \leq m-1) \), compute the total degree \( d_i \), inlinks \( l_{ii} \) and outlinks \( l_{ij} \) between modules \( a_i \) and each of its neighbour modules \( a_j (0 \leq j \leq m-1, i \neq j) \).

### B. CN-specific Genetic Operators

The constraint, that each terminal of a complex network must be present and be present only once in a CN-chromosome, limits the freedom of genetic variation in population. Four CN-specific genetic operations: one-terminal deletion/insertion, two-point terminal exchange, terminal sequence deletion/insertion, and terminal sequence exchange, are designed and developed.

1) **One-terminal deletion/insertion & Terminal Sequence deletion/insertion**

One-terminal deletion/insertion is an operator to transpose a terminal. Two CN-genes: source gene and target gene, are randomly selected from a CN-chromosome. Then a source terminal is randomly picked up from the source gene and transposed into the target gene. To guarantee that each terminal is present once and only once, the source terminal is deleted from its original position after its transposition.

Terminal Sequence deletion/insertion is a generalization of one-terminal deletion/insertion operator. It operates a terminal sequence of random length instead of one terminal.

2) **Two-point terminal exchange & Terminal Sequence Exchange**

Two-point terminal exchange is a variation of two-point mutation in basic GEP. It exchanges two terminals located in two different CN-genes. First, two CN-genes are randomly selected from a CN-chromosome; then one terminal is randomly picked up from each selected CN-gene, respectively; and finally two selected terminals are swapped.

Terminal Sequence Exchange is a generalization of two-point terminal exchange. It exchanges two terminal sequences of the same random length between two randomly selected genes.

### C. Fitness functions

We name the conditional information in the information-theoretic compression approach as **IT-based fitness**, and the modularity is adopted as **modularity-based fitness**.

1) **IT-based fitness**

By following the MDL principle[6, 10], the optimal number of modules \( m \) is the one that minimizes the length of the modular description \( L(Y) \), plus the conditional description length, \( L(X|Y) \), as shown in formula (3) [5].

\[
L(Y) + L(X | Y) = n \log m + \frac{1}{2} m(m+1) \log l + H(Z)
\]

where \( H(Z) = H(X|Y) \) that is the conditional information that the information necessary to describe \( X \) given \( Y \), as defined in formula (4). Given the number of modules \( m \), the best modular description \( Y \) is the one that minimize \( H(Z) \).[5]

\[
H(Z) = \log \left[ \frac{1}{n} \sum_{i=1}^{m} \left( \frac{n_i (n_i - 1) - 1}{2} \right) \prod_{i<j} \left( \frac{n_i n_j}{l_{ij}} \right) \right]
\]  

(4)

2) **Modularity-based fitness**

With *a priori* knowledge how many modules that a network is composed, the modularity \( Q \) [2] is defined as the sum of the contributions from each module \( i \), as shown in formula (5) where \( d_i \) is the total degree of nodes in module \( i \).

The best community structure is found out by maximizing \( Q \).

\[
Q = \sum_{i=0}^{n-1} \frac{l_{ii} - (d_i / 2l)^2}{l_{ii}}
\]

(5)

### D. The CNC-GEP algorithm

Each CN-chromosome is implemented as a linked list containing \( m \) modulation functions and \( n \) node terminals.

To get things started, an initial CN-chromosome is generated as follows: a random list of \( n \) terminals is generated first, and then \( m \) modulation functions: \( M \) are inserted into the randomized terminal list at the positions: \( 0 + i * (n / m) \) (0 ≤ \( i \) ≤ \( m-1 \)).

**Algorithm (CNC-GEP): Complex Network Clustering based-on GEP**

**Input:** Parameters for CNC-GEP, a complex network in Pajak format

**Output:** The best CN-chromosome individual bestInd

1. \( \text{generation} \leftarrow 0; \)
2. \( \text{Generate initial population}; \)
3. while (generation < MaxGeneration || FVCC reaches 100%) {
   4. \( \text{Module Description} Y \leftarrow \text{DecodeChromosome(Ind,)} ; \)
   5. \( \text{Evaluate the fitness of population}; \)
   6. \( \text{Keep the best individual}; \)
   7. \( \text{Select other individuals of population}; \)
   8. \( \text{CN-specific operations at certain rate}; \)
   9. \}
10. \( \text{Output the best individual: bestInd}; \)

### V. Experiments

CNC-GEP is implemented by extending the zGEP source code provided in [11]. Given the number of modules in networks, the performance of CNC-GEP is evaluated by the fraction of vertices classified correctly (FVCC) which is firstly used by Girvan and Newman in [12]. In our experiments, we intend to verify the partitioning generated by CNC-GEP for two benchmark complex networks: the multi-discipline network[13] and Zachary’s karate club network[14]. The classic module structures of these benchmark networks are used as the standard partitioning to evaluate the evolution of individuals. The parameters used per run are summarized in TABLE I and TABLE II.

<table>
<thead>
<tr>
<th>TABLE I. PARAMETER SETTINGS FOR CNC-GEP</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Number of runs</strong></td>
</tr>
<tr>
<td><strong>Function set</strong></td>
</tr>
<tr>
<td><strong>Population size</strong></td>
</tr>
<tr>
<td><strong>Terminal set</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TABLE II. MUTATION RATE SETTINGS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Multi-discipline Network</strong></td>
</tr>
<tr>
<td><strong>Zachary’s karate club network</strong></td>
</tr>
<tr>
<td><strong>Two-point terminal exchange</strong></td>
</tr>
<tr>
<td><strong>Sequence Exchange</strong></td>
</tr>
<tr>
<td><strong>Sequence deletion/insertion</strong></td>
</tr>
</tbody>
</table>

---

1. Algorithm (CNC-GEP): Complex Network Clustering based-on GEP
2. Function set: \( [M] \)
A. Datasets

1) Multi-discipline Network

Multi-discipline network consists of 40 journals as nodes from four different fields: multidisciplinary physics, chemistry, biology, and ecology[13]. The 189 links connect nodes if at least one article from one of the journals cites an article in the other journal during 2004[13]. The network is empirically partitioned into 4 modules with each module having 10 nodes.

2) Zachary’s classic karate club network

Zachary’s classic karate club network is a social network of friendships between 34 members of a karate club at a US university in the 1970 [14]. The 34 nodes connected by 78 edges, and is classically partitioned into 2 modules.

B. Determine the number of modules

By giving the number of modules from 1 to 5, we run CNC-GEP on the both benchmark networks. The best MDL and the best modularity are recorded for each module configuration in the evolution.

Figure 2(a) shows the MDL by partitioning the two networks into 1-5 modules. The lowest MDL is obtained from the 4-module configuration for the multi-discipline network and the 2-module configuration for karate club network, and both results are consistent to what is true in real world.

Figure 2(b) shows the modularity by clustering each of the two networks into 2-5 modules. The maximal modularity obtained in the configurations of 3, 4, and 5 modules are all higher than the one for the classic 2-module configuration. It shows that maximizing the modularity cannot guarantee the detection of the best module structure without the correct number of modules given prior for networks.

C. FVCC for Multi-discipline Network

The statistics of FVCC for 100 runs obtained by running CNC-GEP on multi-discipline network are shown in TABLE III. Figure 3(a) shows the progression of FVCC and Figure 3(b) shows the progression of the best fitness for two successful runs. For these two runs, the best IT-based fitness is obtained by generation of 960, and modularity-based fitness by generation 580.

<table>
<thead>
<tr>
<th>Success rate</th>
<th>Modularity-based(%)</th>
<th>IT-based(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>92.75</td>
<td>81.15</td>
</tr>
</tbody>
</table>

The multi-discipline network is symmetric whose modules are similar in size and in total degree. Evaluating by both the success rate and average FVCC, the modularity-based CNC-GEP outperforms the IT-based CNC-GEP to resolve this symmetric network. As a comparison, Rosvall and Bergstrom can assign 39 of the 40 journals into the proper modules in [5], however they can never 100% correctly resolve the module structure.

D. FVCC for Zachary’s karate club network

The community structures discovered by CNC-GEP based on two different fitness functions are different. The modularity-based CNC-GEP resolves the network as shown in Figure 4(A) in [5], while the information-theory-based...
CNC-GEP partitions nodes with similar roles into two clusters as shown in Figure 4(B) in [5].

<table>
<thead>
<tr>
<th></th>
<th>Modularity-based (%)</th>
<th>IT-based (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Success rate</td>
<td>48</td>
<td>80</td>
</tr>
<tr>
<td>Average FVCC</td>
<td>95.17</td>
<td>99.35</td>
</tr>
</tbody>
</table>

The statistics of FVCC for 100 runs obtained by running CNC-GEP on karate club network are shown in TABLE IV. Figure 4(a) shows the progression of the best FVCC and Figure 4(b) shows the progression of the best fitness for two successful runs. For these two runs, the best IT-based fitness is obtained by generation 80 and the best modularity-based fitness by generation 140.

Zachary’s karate club network is asymmetric in module structure. Both approaches perform well while resolving module structure for this network. Nevertheless, the IT-based CNC-GEP not only assigns nodes to their modules more accurately, but also converges faster than the modularity-based CNC-GEP. The experiment results demonstrate that the IT-based CNC-GEP can resolve the asymmetric network more efficiently and accurately than the modularity-based CNC-GEP.

VI. CONCLUSION

By cohesion superiority of GEP and the information-theoretic compression framework for complex network clustering, we propose a novel algorithm intending to resolve module structures for complex networks. By adopting MDL as the IT-based fitness, CNC-GEP not only determines the optimal number of modules in networks, but also resolves how to cluster nodes into that number of modules by minimizing the mutual information between a network and its modular descriptions so that the best module structure is revealed. A CNC-specific chromosomal organization is designed to encode a module structure for complex networks. Specific genetic operations are created to simulate node moving and exchanging between modules and all these operations always result in valid chromosomes with optimal fitness.

The experiment results on Zachary’s classic karate club network show that only maximizing modularity cannot automatically reveal the optimal module structure if the number of modules in networks is given. With the prior knowledge how many modules compose a network, CNC-GEP based on either the modularity-based or IT-based fitness function performs well. These two approaches may resolve different module structures though. The empirical results show that the IT-based CNC-GEP surpasses the modularity-based CNC-GEP in resolving asymmetric networks, while the modularity-based CNC-GEP is better in discovering module structures for symmetric networks.

REFERENCES